

# Decentralized Algorithms for 3D Symmetric Formations in Robotic Networks: a Contraction Theory Approach

Sumeet Singh, *Student Member, IEEE*, Edward Schmerling, *Student Member, IEEE*, Marco Pavone, *Member, IEEE*

**Abstract**—This paper presents decentralized algorithms for formation control of multiple robots in three dimensions. Specifically, we leverage the mathematical properties of cyclic pursuit along with results from contraction and partial contraction theory to design decentralized control algorithms that ensure global convergence to symmetric formations. We first consider regular polygon formations as a base case, and then extend the results to Johnson solid and other polygonal mesh formations. The algorithms are further augmented to allow control over formation size and avoid collisions with other robots in the formation. The robustness properties of the algorithms are assessed in the presence of bounded additive disturbances and their effect on the *quality* of the formation is quantified. Finally, we present a general methodology for embedding the control laws on complex dynamical systems, in this case, quadcopters, and validate this approach via simulations and experiments on a fleet of quadcopters.

## I. INTRODUCTION

Recently there has been much interest in decentralized, multi-robot systems due to the plethora of possible applications and performance advantages. These systems have many attractive properties such as robustness to single-point failures, scalable implementation, and potentially lower operating costs when compared with monolithic approaches. Furthermore, multiple robots can together accomplish tasks which may be impossible or very difficult for any single robot on its own [1]–[3].

A prototypical problem in this context is decentralized formation control, that is, the problem of ensuring convergence to a desired formation via control algorithms amenable to a distributed implementation [4]–[6]. Most existing results consider two-dimensional formations [7]–[9], however, a number of robotic applications would require three-dimensional (3D) formations. For example, it may be desired to deploy a set of small satellites to surround a larger damaged spacecraft to provide complete 3D reconstruction and visualization of the external structure. Additional examples include differential atmospheric, deep space, or underwater measurements of environmental phenomena, and filming sport and entertainment events [10]–[12].

Accordingly, the objective of this paper is to design and rigorously analyze decentralized control algorithms for 3D formations. Our approach leverages the simple, yet effective strategy of cyclic pursuit. Essentially, the cyclic pursuit strategy entails letting each robot  $i$  follow its leading neighbor

$i + 1$  modulo  $n$ , where  $n$  is the number of robots. This approach is attractive due to its decentralized nature and low information requirements, namely relative position, and has led to its popularity in recent years [13]–[15]. In this paper, we adopt the cyclic pursuit strategy proposed in [16] (that, in turn, generalizes earlier results in [14], [15]), where each robot follows its neighbors on both sides, that is, robots  $i + m$  and  $i - m$ , where  $m = \{1, 2, \dots, N\}$  is a look-ahead parameter up to a horizon  $N$ . This is referred to as the *symmetric* cyclic control algorithm. To prove stability (i.e., convergence to a desired formation), we exploit contraction theory. Contraction theory is a relatively recent innovation in control system design [17], and hinges upon an exact differential analysis of convergence. At its core, contraction theory studies exponential convergence of pairs of system trajectories towards each other, and by extension, to a desired target trajectory. This represents a generalization with respect to traditional Lyapunov analysis which studies convergence to the origin, i.e., to a zero trajectory [18]. Partial contraction theory extends this concept by considering the convergence of trajectories to a “set of properties,” for example, a flow-invariant subspace [19], [20].

The contributions of this paper are as follows. First, we study convergence of the symmetric cyclic control algorithm to a polygon formation in 3D. Our analysis differs from the analysis in [16] in that it considers a refined notion of the target formation (necessary to properly constrain a polygon formation). Our analysis (as in [16]) leverages partial contraction theory and, in addition, yields insights into the various tuning parameters for the controller and their impact on performance metrics such as convergence rate and control saturation. Second, we extend the definition of the target formation to accommodate combinations of regular polygons such as the set of convex regular polyhedra known as Johnson solids, as well as more general polygonal mesh formations. Control laws and sufficient conditions for convergence are derived for these relatively complex polyhedron formations, paving the way for innovative mission concepts [21]. Third, we provide extensions to the algorithm to control the size of the formation, and prevent collisions with other robots. Fourth, we analyze the robustness properties of the symmetric cyclic controller under bounded additive disturbances and provide performance bounds with respect to the formation error. These results provide theoretical insights into experiments performed on board the International Space Station which highlighted the remarkable robustness properties of cyclic controllers [15]. Finally, we present a control hierarchy for implementing the formation control laws on a complex dynamical system, in this case, a fleet of quadcopters.

The rest of the paper proceeds as follows: Section II

Sumeet Singh and Marco Pavone are with the Department of Aeronautics and Astronautics; Edward Schmerling is with the Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, 94305, {ssingh19, schmrlng, pavone}@stanford.edu.

This work was supported in part by the Stanford Graduate Fellowship (SGF) and the King Abdulaziz City for Science and Technology (KACST).

introduces the mathematical background for contraction and partial contraction theory and circulant matrices which forms the basis for all subsequent analysis. Section III formalizes the problem, introduces the formation subspace and symmetric cyclic control law, and provides conditions for convergence to a regular polygon. In Section IV, we extend the results to polyhedral formations. Section V provides extensions to the algorithm for controlling formation size and preventing collisions among the robots. We also assess the robustness properties of the symmetric cyclic controller under bounded additive disturbances. In Section VI, we demonstrate the application of our formation control laws on quadcopters, providing verification via simulations and hardware experiments. Finally, in Section VII, we draw our conclusions and provide avenues for future work.

A preliminary version of this paper appeared as [22]. This extended and revised version contains as additional contributions: (1) extensions to the control algorithms for forming more complex polygonal mesh formations, controlling formation size, and avoiding collisions, (2) additional numerical results, and (3) hardware experimental results on a fleet of quadcopters.

## II. MATHEMATIC PRELIMINARIES

### A. Notation

Given a square matrix  $A$ , the symmetric part of  $A$ , i.e.,  $(1/2)(A + A^T)$ , is denoted by  $A_{\text{sym}}$ . The smallest and largest eigenvalues of  $A_{\text{sym}}$  are denoted, respectively, by  $\lambda_{\min}(A)$  and  $\lambda_{\max}(A)$ . Accordingly, the matrix  $A$  is positive definite (denoted  $A \succ 0$ ) if  $\lambda_{\min}(A) > 0$ , and negative definite (denoted  $A \prec 0$ ) if  $\lambda_{\max}(A) < 0$ . Let  $I_k$  denote the  $k$ -by- $k$  identity matrix. Let  $A \otimes B$  denote the Kronecker product between matrices  $A$  and  $B$ . The null space of a matrix  $A$  is denoted by  $\mathcal{N}(A)$ . Finally, the set of eigenvalues of  $A$  is denoted by  $\lambda(A)$ .

### B. Contraction and Partial Contraction Theory

This section reviews basic results in nonlinear contraction theory [17], [23]. The basic stability result in contraction theory reads as follows.

**Theorem II.1** (Contraction [17]). *Consider a general system of the form*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad (1)$$

where  $\mathbf{x}$  is the  $n \times 1$  state vector and  $\mathbf{f}$  is a  $n \times 1$  nonlinear, continuously differentiable vector function with Jacobian  $\partial \mathbf{f} / \partial \mathbf{x}$ . If there exists a square matrix  $\Theta(\mathbf{x}, t)$  such that  $\Theta(\mathbf{x}, t)^T \Theta(\mathbf{x}, t)$  is uniformly positive definite and the matrix

$$F := \left( \dot{\Theta} + \Theta \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right) \Theta^{-1}$$

is uniformly negative definite, then all system trajectories converge exponentially to a single trajectory. In this case, the system is said to be contracting.

A few remarks are in order. First, a matrix  $\Theta(\mathbf{x}, t)$  is uniformly positive definite if there exists  $\beta > 0$  such that  $\forall(\mathbf{x}, t)$ ,  $\lambda_{\min}(\Theta(\mathbf{x}, t)) \geq \beta$ , and is uniformly negative definite if there exists  $\beta > 0$  such that  $\forall(\mathbf{x}, t)$ ,  $\lambda_{\max}(\Theta(\mathbf{x}, t)) \leq -\beta$ . Second, the matrix  $F$  is referred to as the *generalized Jacobian*

for system (1). Third, the matrix  $M(\mathbf{x}, t) := \Theta(\mathbf{x}, t)^T \Theta(\mathbf{x}, t)$  is referred to as the *contraction metric*.

We next discuss partial contraction theory, which allows one to address questions more general than trajectory convergence. Consider system (1), and assume there exists a flow-invariant linear subspace  $\mathcal{M} \subset \mathbb{R}^n$ , i.e., a linear subspace with the property that, for all  $t$ ,

$$\mathbf{x} \in \mathcal{M} \Rightarrow \mathbf{f}(\mathbf{x}, t) \in \mathcal{M}.$$

Assume that the dimension of  $\mathcal{M}$  is  $p$  and let  $(\mathbf{e}_1, \dots, \mathbf{e}_n)$  be an orthonormal basis of  $\mathbb{R}^n$  where the first  $p$  vectors form a basis for  $\mathcal{M}$ . Let  $\bar{U}$  be a  $p \times n$  matrix whose rows are  $\mathbf{e}_1^T, \dots, \mathbf{e}_p^T$ . Let  $\bar{V}$  be an  $(n - p) \times n$  matrix whose rows are  $\mathbf{e}_{p+1}^T, \dots, \mathbf{e}_n^T$ . One can easily verify that matrix  $\bar{V}$  is sub-unitary and satisfies the properties  $\bar{V}^T \bar{V} + \bar{U}^T \bar{U} = I_n$ ,  $\bar{V} \bar{V}^T = I_{n-p}$ , and  $\mathbf{x} \in \mathcal{M}$  if and only if  $\bar{V} \mathbf{x} = \mathbf{0}$ . We will refer to  $\bar{V}$  as the projection matrix of  $\mathcal{M}$ . The main theorem in partial contraction theory can be stated as:

**Theorem II.2** (Partial Contraction [20]). *Consider a flow-invariant linear subspace  $\mathcal{M}$  and its associated projection matrix  $\bar{V}$ . A particular solution  $\mathbf{x}_p(t)$  of system (1) converges exponentially to  $\mathcal{M}$  if the auxiliary system*

$$\dot{\mathbf{y}} = \bar{V} \mathbf{f}(\bar{V}^T \mathbf{y} + \bar{U}^T \bar{U} \mathbf{x}_p(t), t)$$

is contracting with respect to  $\mathbf{y}$ . If this is true for all particular solutions  $\mathbf{x}_p$ , all trajectories of system (1) will exponentially converge to  $\mathcal{M}$  from all initial conditions.

Combining Theorems II.1 and II.2, one obtains a powerful tool to prove convergence to a desired flow-invariant subspace.

**Corollary II.3** (Convergence to Flow-invariant Subspace [20]). *A sufficient condition for global exponential convergence to  $\mathcal{M}$  is*

$$\bar{V} \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \bar{V}^T \prec 0, \quad \text{uniformly.}$$

In this paper, the subspace  $\mathcal{M}$  will represent a desired robot formation.

**Remark II.4** (Non-orthonormal  $V$ ). *Note that the application of partial contraction theory requires the rows of the projection matrix  $\bar{V}$  to be orthonormal. However, the matrix  $V$  characterizing a subspace  $\mathcal{M}$  may not be row-wise orthonormal, e.g., when it is obtained by combining a set of linearly independent equations. Nonetheless, as long as the equations are independent, the matrix  $V$  will be full row rank and can be transformed via an invertible matrix  $T$  into an orthonormal counterpart  $\bar{V}$  which satisfies (1)  $V = T \bar{V}$ , (2)  $\bar{V} \mathbf{x} = \mathbf{0} \Leftrightarrow \mathbf{x} \in \mathcal{M}$ , and (3)  $\bar{V} \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \bar{V}^T \prec 0 \Leftrightarrow V \frac{\partial \mathbf{f}}{\partial \mathbf{x}} V^T \prec 0$  [20].*

### C. Circulant and Block-Circulant Rotational Matrices

A circulant matrix of order  $n$  is a square matrix with the following structure:

$$C = \begin{bmatrix} c_1 & c_2 & \cdots & c_n \\ c_n & c_1 & \cdots & c_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_2 & c_3 & \cdots & c_1 \end{bmatrix}. \quad (2)$$

The elements of each row are identical to the row above, but shifted one position to the right and wrapped around. Thus, a circulant matrix can be compactly denoted as:

$$C = \text{circ}[c_1 \ c_2 \ \dots \ c_n].$$

A useful circulant matrix with dimensions  $n \times n$  used in this paper is defined below, parametrized by the integer  $m \in \{0, \dots, n-1\}$ :

$$L_m := \text{circ}[1, 0, \dots, 0, \underbrace{-1}_{(m+1)\text{st element}}, 0, \dots, 0].$$

#### D. Kronecker Product

Let  $A$  and  $B$  be matrices of size  $m \times n$  and  $p \times q$  respectively. The Kronecker product of  $A$  and  $B$  is defined as:

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix},$$

where  $A \otimes B$  has dimensions  $mp \times nq$ . If  $\lambda_A$  is an eigenvalue of  $A$  with eigenvector  $\mathbf{v}_A$ , and similarly  $\lambda_B$  and  $\mathbf{v}_B$  are an eigenvalue and eigenvector pair for  $B$  then the corresponding eigenvalue and eigenvector pair for  $A \otimes B$  is  $\lambda_A \lambda_B$  and  $\mathbf{v}_A \otimes \mathbf{v}_B$ . We also note the following identity:  $(A \otimes B)(C \otimes D) = AC \otimes BD$ , where  $A, B, C$  and  $D$  have compatible dimensions in order for the products  $AC$  and  $BD$  to be well defined.

In the following, we will denote by  $\mathcal{C}_{L \otimes R}$  the set of block-circulant matrices that can be written as  $L \otimes R_\beta$ , where  $L$  is a circulant matrix and  $R_\beta$  is a rotation matrix about the axis  $\mathbf{e}_z := (0, 0, 1)^T$  with rotation angle  $\beta$ . From the properties of the Kronecker product, the eigenvalues and eigenvectors of a matrix in  $\mathcal{C}_{L \otimes R}$  are given, respectively, by the product of eigenvalues and Kronecker product of eigenvectors of the circulant matrix  $L$  and the rotation matrix  $R_\beta$ .

### III. SYMMETRIC PLANAR FORMATIONS

#### A. Problem Setup

Consider  $n \geq 3$  mobile robots (uniquely labelled by an integer  $i \in \{1, \dots, n\}$ ). Denote the position of robot  $i$  at time  $t$  as  $\mathbf{x}_i(t)$ , where  $\mathbf{x}_i(t) \in \mathbb{R}^3$ . The overall state vector is denoted by  $\mathbf{x} = (\mathbf{x}_1^T, \mathbf{x}_2^T, \dots, \mathbf{x}_n^T)^T$ . The dynamics of each robot are given by

$$\dot{\mathbf{x}}_i = \mathbf{g}_i(\mathbf{x}) + \mathbf{u}_i(\mathbf{x}, t), \quad (3)$$

where  $\mathbf{u}_i$  is the control action. It is desired to design the control actions so that (a) they drive the global state vector  $\mathbf{x}$  to a desired symmetric formation, and (b) they are amenable to a decentralized implementation. As in [16], our strategy is to “encode” a symmetric formation as a formation subspace as discussed next. The proofs for all theorems and lemmas introduced in this section are provided in the appendix.

#### B. Formation Subspace

In this section we consider *regular polygons* in 3D as desired symmetric formations – the extension to non-planar formations (which represents the main contribution of this paper) is discussed in Section IV. Consider the case where the direction normal to the desired formation polygon is aligned with the vector  $\mathbf{e}_z$  (the general case can be reduced to this case

via a coordinate transformation). We encode such a formation via the subspace:

$$\mathcal{M}_n = \{\mathbf{x} \in \mathbb{R}^{3n} :$$

$$(\mathbf{x}_{i+1} - \mathbf{x}_i) = R_{2\pi/n}(\mathbf{x}_{i+2} - \mathbf{x}_{i+1}), i = 1, \dots, n-2, \quad (4a)$$

$$\mathbf{e}_z^T(\mathbf{x}_n - \mathbf{x}_{n-1}) = \mathbf{e}_z^T(\mathbf{x}_1 - \mathbf{x}_n)\}, \quad (4b)$$

where the indices are considered modulo  $n$ , and  $R_{2\pi/n}$  denotes a counterclockwise rotation around  $\mathbf{e}_z$ . The  $n-2$  constraints in (4a) will be referred to as *rotational constraints* while the single constraint in (4b) will be referred to as the *in-plane* constraint. The in-plane constraint, not considered in [16], is needed in order to ensure that all robots lie in the same plane (which is not ensured by the rotational constraints alone as they permit spiral formations). The following lemma shows that the constraints (4a) and (4b) are indeed necessary and sufficient for the definition of a regular polygon.

**Lemma III.1** (Polygon Constraints). *The set of constraints (4a) and (4b) are necessary and sufficient for the definition of a regular polygon with normal direction  $\mathbf{e}_z$ .*

**Remark III.2** (Polygon Degrees of Freedom). *The constraints (4a) and (4b) together form a set of  $3n-5$  linearly independent equations in  $3n$  variables. The five missing equations correspond to five distinct degrees of freedom: three in translation, one in scaling, and one in in-plane rotation. That is, the polygon may be translated anywhere in space, scaled in size, or rotated within the desired plane about the plane normal.*

Both (4a) and (4b) can be compactly represented as the null space of a certain matrix, as shown in the following lemma.

**Lemma III.3** (Compact Constraints). *Let  $W_{r_n} := [I_{n-2}, 0_{(n-2) \times 2}]$  and  $W_{p_n} := [0_{1 \times (n-2)}, 1, 0]$ . Define the  $3(n-2) + 1 \times 3n$  matrix*

$$V := \underbrace{\begin{bmatrix} W_{r_n} \otimes I_3 \\ W_{p_n} \otimes \mathbf{e}_z^T \end{bmatrix}}_{:= \mathcal{W}_n} \underbrace{(L_1 \otimes I_3 + (L_1 - L_2) \otimes R_{2\pi/n})}_{:= \mathcal{P}_n \in \mathcal{C}_{L \otimes R}}. \quad (5)$$

*Then the rotational and in-plane constraints in equation (4a) and (4b) are equivalent to the equation  $V\mathbf{x} = 0$ .*

Note that  $W_{r_n}$  captures the rotational constraints, while  $W_{p_n}$  captures the in-plane constraint. The subspace  $\mathcal{M}_n$  can then be characterized as the null space of matrix  $V$ , i.e.,  $V\mathbf{x} = \mathbf{0} \Leftrightarrow \mathbf{x} \in \mathcal{M}_n$ .

Our standing assumption throughout this paper is that the robots’ internal dynamics, i.e.,  $\mathbf{g}(\mathbf{x})$ , are flow-invariant with respect to  $\mathcal{M}_n$ :

**Assumption 1** (Flow-invariance). *The internal dynamics are flow-invariant with respect to the desired formation, that is for all  $\mathbf{x} \in \mathcal{M}_n$  one has  $V\mathbf{g}(\mathbf{x}) = \mathbf{0}$ .*

#### C. Symmetric Cyclic Controller

We consider the class of symmetric cyclic controllers proposed in [16] (in turn generalizing the pursuit controllers introduced in [14]):

$$\mathbf{u}_i = \sum_{m=1}^N k_m [R_m(\mathbf{x}_{i+m} - \mathbf{x}_i) + R_m^T(\mathbf{x}_{i-m} - \mathbf{x}_i)], \quad (6)$$

where  $N$  is the look-ahead horizon ( $0 < N < n-1$ ),  $k_m > 0$  is a gain,  $R_m$  is a rotation matrix around  $\mathbf{e}_z$  with rotation angle

$\alpha_m$ , and  $(\mathbf{x}_{i+m} - \mathbf{x}_i)$  and  $(\mathbf{x}_{i-m} - \mathbf{x}_i)$  denote the relative coordinates among robot  $i$  and its  $i+m$  and  $i-m$  neighbors, where  $m = \{1, \dots, N\}$  and summation is modulo  $n$ . Note that the control law is spatially-decentralized, as each robot only requires relative position information from a set of neighboring robots.

Then, the symmetric cyclic controller can be written in compact form as

$$\begin{aligned} \mathbf{u} &= - \sum_{m=1}^N k_m [L_m \otimes R_m + L_m^T \otimes R_m^T] \mathbf{x} \\ &= - \sum_{m=1}^N k_m \mathcal{L}_m \mathbf{x} = -\mathcal{L} \mathbf{x}, \end{aligned} \quad (7)$$

where  $\mathcal{L}_m := L_m \otimes R_m + L_m^T \otimes R_m^T \in \mathcal{C}_{L \otimes R}$  and  $\mathcal{L} := \sum_{m=1}^N k_m \mathcal{L}_m$ . In order to apply partial contraction theory, the control law needs to be flow-invariant (note that, by Assumption 1, the internal dynamics are flow invariant). The next lemma shows that this is indeed the case.

**Lemma III.4** (Flow-invariance). *Subspace  $\mathcal{M}_n$  is flow-invariant with respect to the symmetric cyclic control law given in equation (7).*

We are now in a position to apply partial contraction theory to show that, under some assumptions, the symmetric cyclic controller drives the system to the desired formation subspace  $\mathcal{M}_n$ .

As the encoded constraints are linearly independent,  $V$  is full row rank, guaranteeing the existence of an orthonormal counterpart  $\bar{V}$  (see Remark II.4). In the following, to prove convergence to  $\mathcal{M}_n$ , we will apply partial contraction theory using  $\bar{V}$ . Let  $\bar{U}$  be a matrix whose rows represent an orthonormal basis for the orthogonal complement of the subspace defined by the rows of  $\bar{V}$ . According to Theorem II.2, we want to show that for system

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}) - \mathcal{L} \mathbf{x},$$

the associated auxiliary system

$$\dot{\mathbf{y}} = \bar{V} \left( \mathbf{g}(\bar{V}^T \mathbf{y} + \bar{U}^T \bar{U} \mathbf{x}_p) - \mathcal{L} (\bar{V}^T \mathbf{y} + \bar{U}^T \bar{U} \mathbf{x}_p) \right),$$

is contracting. Note that by Assumption 1 and Lemma III.4 the closed-loop dynamics are invariant with respect to  $\mathcal{M}_n$ . Then, according to Corollary II.3, one requires

$$\bar{V} \left( \frac{\partial \mathbf{g}}{\partial \mathbf{x}} - \mathcal{L} \right) \bar{V}^T \prec 0, \quad \text{uniformly.}$$

By Remark II.4, since  $V$  and  $\bar{V}$  are related by an invertible transformation, the above stability requirement can be reformulated in terms of  $V$  directly. By factoring  $V$  as  $\mathcal{W}_n \mathcal{P}_n$  (see equation (5)) one obtains the condition

$$\mathcal{W}_n \mathcal{P}_n \left( \frac{\partial \mathbf{g}}{\partial \mathbf{x}} - \mathcal{L} \right) \mathcal{P}_n^T \mathcal{W}_n^T \prec 0, \quad \text{uniformly.} \quad (8)$$

Performing an eigenvalue analysis of (8) yields the main result of this section. The details of the proof are provided in the appendix.

**Theorem III.5** (Polygon Convergence). *Assume*

$$\sup_{\mathbf{x}, t} \left( \lambda_{\max} \left( \mathcal{P}_n \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \mathcal{P}_n^T \right) - \min_{\substack{1 \leq i \leq n \\ k \in \{-1, 0, 1\}}} \sum_{m=1}^N k_m \lambda_{ik}^{(m)} \right) < 0, \quad (9)$$

where

$$\lambda_{ik}^{(m)} = \frac{2}{e^{\frac{2\pi(2(i-1)+k)j}{n}}} \left[ \cos(k\alpha_m) - \cos \left( k\alpha_m + \frac{2\pi m(i-1)}{n} \right) \right] \left[ \left( e^{\frac{2\pi(i-1+k)j}{n}} - 1 \right) \left( e^{\frac{2\pi(i-1)j}{n}} - 1 \right) \right]^2.$$

*Then the robots governed by system (3) under the symmetric cyclic controller (6) globally converge to a regular polygon formation, i.e., to the formation subspace  $\mathcal{M}_n$ .*

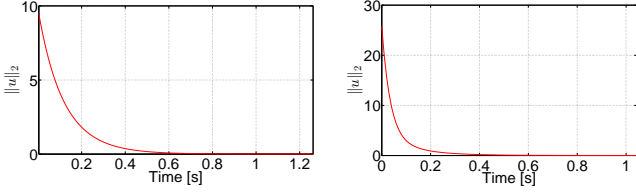
Note that due to the necessary inclusion of the in-plane constraint, the eigenvalues in Theorem III.5 differ from those in [16].

**Remark III.6** (Formations of Fixed Size). *Theorem III.5 provides a sufficient condition to ensure that the robots converge to a regular polygon formation. Note that depending on the internal dynamics  $\mathbf{g}(\mathbf{x})$  and the rotation angle  $\alpha_m$ , the polygon formation will be contracting, expanding, or of fixed size. For instance, if  $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ , by setting  $\alpha_m < m\pi/n$  the polygon formation will contract towards a point; by setting  $\alpha_m > m\pi/n$  the polygon formation will expand as the robots travel along rays emanating from the center of mass of the formation; by setting  $\alpha_m = m\pi/n$  the polygon formation will be of fixed size, where the size depends on the initial conditions. The freedom in scaling the formation will be utilized later to control the formation size.*

For a given number of robots, the primary design parameters in the symmetric cyclic controller include the gains  $k_m$  and the look-ahead horizon  $N$ . It is clear that increasing the gains uniformly scales all eigenvalues of the projected Jacobian, thereby increasing the exponential convergence rate. However, this is at the expense of a more aggressive controller which increases the risk of control saturation. On the other hand, increasing the look-ahead horizon, while imposing greater information requirements for each robot, also increases the convergence rate but at a lower risk of control saturation. This is due to the fact that the symmetric cyclic controller leverages the degree of asymmetry at each node of the polygon. Increasing the look-ahead horizon therefore improves the “estimate” of this asymmetry, rather than simply magnifying the control law by increasing the gain. This intuition is exemplified by the following simulation.

For simplicity, we assume that the internal dynamics  $\mathbf{g}(\mathbf{x})$  are zero. In order to compare the control signals due to varying choices in gains and look-ahead, we maintain the same lower bound on the closed-loop convergence rate, i.e.,  $\lambda_{\min}(\bar{V} \mathcal{L} \bar{V})$ . Figure 1 shows the Euclidean norm of the net control signal, i.e.,  $\|\mathcal{L} \mathbf{x}(t)\|_2$  for six robots starting at the same initial conditions. The control signal in Figure 1a demonstrates a significantly lower peak value than the signal in Figure 1b despite the fact that the two closed-loop systems possess the same lower bound on convergence rate.

**Remark III.7** (General Plane of Convergence). *For polygons not orthogonal to  $\mathbf{e}_z$ , one can use a similarity transformed version of the control rotation matrix  $R_m$  to obtain analogous convergence results. Let  $R_\eta^T \mathbf{e}_z$  be the desired polygon normal*



(a) Control norm with  $N = 2$ ,  $k_m = 2$ . (b) Control norm with  $N = 1$ ,  $k_m = 6.928$ .

Fig. 1: Comparison of control effort with varying gains and look-ahead horizon.

where  $R_\eta$  is an arbitrary rotation matrix. Define  $R_{m_s} := R_\eta^T R_m R_\eta$  to be the similarity transform of  $R_m$ . We now replace  $R_m$  in the individual control law given in (6) by  $R_{m_s}$ , and the expression in (7) becomes:

$$\begin{aligned} \mathbf{u} &= - \underbrace{(I_n \otimes R_\eta^T)}_{:= \mathcal{R}_\eta^T} \sum_{m=1}^N k_m \mathcal{L}_m \underbrace{(I_n \otimes R_\eta)}_{:= \mathcal{R}_\eta} \mathbf{x} \\ &= - \sum_{m=1}^N k_m \mathcal{L}_{m\eta} \mathbf{x} = - \mathcal{L}_\eta \mathbf{x}, \end{aligned} \quad (10)$$

where  $\mathcal{L}_{m\eta} := \mathcal{R}_\eta^T \mathcal{L}_m \mathcal{R}_\eta$  and  $\mathcal{L}_\eta := \sum_{m=1}^N k_m \mathcal{L}_{m\eta}$ . The expression for the projection matrix  $V$  becomes:

$$V = \mathcal{W}_n \mathcal{P}_n \mathcal{R}_\eta. \quad (11)$$

The convergence analysis under control law (10) is virtually identical to the one provided above for the case where the polygon is orthogonal to  $\mathbf{e}_z$ . The generalized control law (10) will be leveraged in Section IV to design control laws for polyhedral formations.

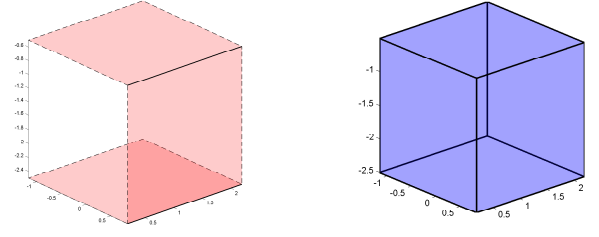
#### IV. POLYHEDRAL FORMATIONS

In this section we extend the results of Section III to polyhedral formations. Specifically, we first focus on convex polyhedra having regular faces and equal edge lengths, referred to as Johnson solids. Generalizations to other polygonal mesh formations are discussed at the end of this section.

**Definition IV.1** (Convex Polyhedral Solid and Polyhedral Surface). A convex solid polyhedron, or simply polyhedron, is defined as a body bounded by a finite number of polygons such that it lies on one side of the plane of each polygon. Equivalently, it is the convex hull of the polygon vertices. A complete polyhedral surface is the set of bounding faces for a polyhedron. A partial polyhedral surface (PPS) is a subset of the faces of a complete polyhedral surface.

Figure 2 illustrates the distinction between a complete and partial surface of a convex solid in the definition above. In Figure 2a, only a portion of the surface of the Johnson solid (cube) is specified, and thus the PPS formed by the faces in Figure 2a contains un-matched edges. The set of these unmatched edges (shown in dashed form in Figure 2a) is termed the *boundary* of the PPS. The complete polyhedral surface displayed in Figure 2b encompasses the entire surface of the solid cube.

Consider now a polyhedron  $\mathcal{P}$  with complete polyhedral surface  $\mathcal{F}$ . As an extension to the discussion in Section III, each



(a) Partial polyhedral surface of a cube. (b) Complete polyhedral surface of a cube.

Fig. 2: Partial and complete representations of a bounded polyhedron (cube) with same vertex set.

face of  $\mathcal{F}$  can be described by a set of linearly independent constraints on its vertices. When considering these faces in combination, we must ensure that the full set of constraints (i.e., due to all faces in  $\mathcal{F}$ ) remain linearly independent. This motivates the search for a PPS representation of the polyhedron that is *minimal* in the sense that it does not induce redundant constraints yet still retains enough information to fix the positions of each vertex (robot). We note that this reduced set of linear constraints for the PPS will retain the translational and scaling degrees of freedom associated with the PPS's constituent polygons (see Remark III.2). Thus all further discussion of the terms “face” or “polygon” in this section will implicitly assume translation and scale invariance. In the following subsection, we show that a suitable minimal PPS representation can always be constructed for any given convex polyhedron.

##### A. Formulation of a Minimal Partial Polyhedral Surface

We first introduce some terminology (adapted from [24]):

**Definition IV.2** (Development of a Polyhedral Surface [24]). A development  $\mathcal{D}$  is defined as the tuple  $(\mathcal{Q}, \mathcal{R})$  where  $\mathcal{Q}$  is a set of polygons as defined in Section III and  $\mathcal{R}$  is a set of rules that identify common edges between polygons. The sets  $\mathcal{Q}$  and  $\mathcal{R}$  satisfy the following properties:

- 1) For every pair of adjacent polygons  $\mathcal{Q}_i, \mathcal{Q}_j \in \mathcal{Q}$ , there is a rule  $r_{ij} \in \mathcal{R}$  which encompasses two pieces of information, namely, it identifies (a) the two shared vertices between the two polygons, and (b) the dihedral angle formed between the planes of the two polygons.
- 2) It is possible to create a path between any two polygons in the collection by following the common edges (i.e., a development cannot be split into disconnected parts).
- 3) Each side of every polygon is either matched with no edge of any other polygon, or with exactly one edge of another polygon in the collection.

Every development defines a *unique* (complete or partial) polyhedral surface  $\mathcal{F}$ , but multiple developments may correspond to the same surface. The surface in Figure 2a corresponds to a development with three squares, while the surface in Figure 2b corresponds to a development with six squares (this development is commonly referred to as the polyhedral net [25]). Our goal is to be able to use the development for the PPS in Figure 2a as a viable representation of the complete polyhedral surface in Figure 2b by leveraging the following result:

**Lemma IV.3** (Completion of a Polyhedral Surface [24]). *Every bounded convex PPS  $\mathcal{F}'$  gives rise to a unique complete polyhedral surface  $\mathcal{F}$  without adding new vertices. If  $\mathcal{A}'$  denotes the set of vertices for  $\mathcal{F}'$ , then the completion  $\mathcal{F}$  is given by the surface of the convex hull of  $\mathcal{A}'$ .*

We are now ready to present the first step in constructing a PPS representation of a complete polyhedral surface where the set of linear constraints induced by the faces of the PPS are linearly independent.

**Lemma IV.4** (Minimal Development). *Let  $\mathcal{F}$  be a complete polyhedral surface and let  $\mathcal{A}$  denote its set of vertices. Suppose  $\mathcal{D} = (\mathcal{Q}, \mathcal{R})$  is a development for  $\mathcal{F}$ . Choose a set of polygons  $\mathcal{Q}' \subset \mathcal{Q}$  that satisfies the following properties:*

- 1) *Tree: Within the dual graph of  $\mathcal{F}$ , the subgraph induced by  $\mathcal{Q}'$  is a tree,*
- 2) *Vertex span: The set of vertices spanned by  $\mathcal{Q}'$  is identical to the set of vertices spanned by  $\mathcal{Q}$ .*

*Given  $\mathcal{Q}'$  satisfying the properties above, we now construct the set of rules  $\mathcal{R}' \subset \mathcal{R}$  corresponding to the polygons selected in  $\mathcal{Q}'$ :*

$$\mathcal{R}' = \{r_{ij} \in \mathcal{R} \mid Q_i, Q_j \in \mathcal{Q}'\}.$$

*Then, the tuple  $\mathcal{D}' := (\mathcal{Q}', \mathcal{R}')$  is a well-posed development (essentially a spanning tree of the graph  $\mathcal{D}$ ) and therefore defines a PPS  $\mathcal{F}'$ . Furthermore, the completion of  $\mathcal{F}'$  is the original surface  $\mathcal{F}$ .*

*Proof.* Properties 1) and 3) of Definition IV.2 hold for  $\mathcal{Q}', \mathcal{R}'$  by construction. Property 2) holds because the ‘tree’ assumption for  $\mathcal{Q}'$  ensures that there exists a path between any two polygons in  $\mathcal{Q}'$  by following common edges. Thus  $\mathcal{D}'$  is well-defined and therefore describes a PPS  $\mathcal{F}'$ .

The ‘vertex span’ assumption ensures that the convex hulls of the vertices of  $\mathcal{F}$  and  $\mathcal{F}'$  are identical. Since  $\mathcal{F}$ , a complete polyhedral surface, is the surface of this convex hull, by Lemma IV.3 the completion of  $\mathcal{F}'$  is  $\mathcal{F}$ .  $\square$

The key message of Lemma IV.4 is that a PPS is sufficient to describe a polyhedron (i.e., the desired formation) provided that it encompasses all vertices of the polyhedron and its faces are connected. More precisely, each face within the PPS defines a set of linear constraints on its vertices, the combination of which completely determine the convex hull matching the desired formation. The further restriction that the faces induce a tree graph is an important step, as shown next, towards ensuring that the combined linear constraints are independent. For the rest of this section, we assume that a PPS  $\mathcal{F}$  described by the development  $\mathcal{D} = (\mathcal{Q}, \mathcal{R})$  has been fixed. Denote  $L := |\mathcal{Q}|$ , the number of polygons in  $\mathcal{F}$ . We now make explicit the notion of rules of a development.

Let  $Q_k$  denote the  $k^{\text{th}}$  polygon in  $\mathcal{Q}$ . Each  $Q_k$  is associated with a set of robots  $\mathcal{V}_k$ , and a unit outward normal  $\mathbf{n}_k$  (pointing outward from the interior of the polyhedron). The number of robots  $n$  is equal to the number of vertices in the polyhedron and each robot is uniquely associated with a vertex. Each face induces a counterclockwise (with respect to  $\mathbf{n}_k$ ) sub-indexing of the robots, specifically, let  $i_k$  denote the index of robot  $i$  in face  $k$ , where  $i = 1, \dots, n$  and  $k = 1, \dots, L$ , see Figure 3. If robot  $i$  does not belong to face  $k$ , then we use the convention  $i_k = \infty$ .

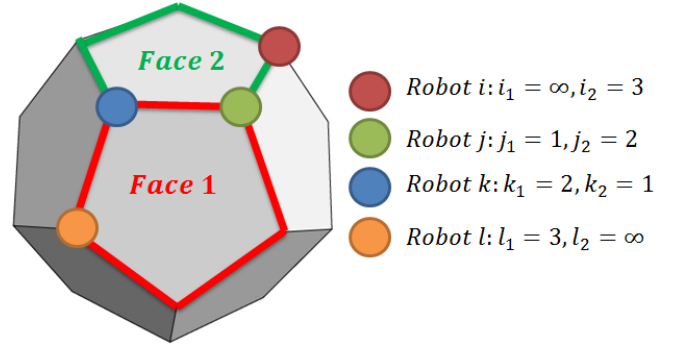


Fig. 3: Indexing of robots for a general polyhedron.

For each  $Q_k \in \mathcal{Q}$ , define the matrix  $\bar{E}^{(k)} \in \mathbb{R}^{|\mathcal{V}_k| \times n}$  as

$$[\bar{E}^{(k)}]_{ij} = \begin{cases} 1 & \text{if } j_k = i, \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

Thus  $[\bar{E}^{(k)}]_{ij}$  equals one if robot  $j$ 's index within face  $k$  is equal to  $i$ . Define  $\mathbf{x}^{(k)} \in \mathbb{R}^{3|\mathcal{V}_k|}$  to be the positions of the robots in face  $Q_k$ . Using (12), we may write  $\mathbf{x}^{(k)}$  as

$$\mathbf{x}^{(k)} = (\bar{E}^{(k)} \otimes I_3) \mathbf{x} = E^{(k)} \mathbf{x}, \quad (13)$$

where  $\mathbf{x} \in \mathbb{R}^{3n}$  is the global state vector, and  $E^{(k)} := \bar{E}^{(k)} \otimes I_3$ . As a direct extension of Section III-B, the face  $Q_k$  (by assumption a regular polygon) is defined as the subspace  $\mathcal{M}_{|\mathcal{V}_k|}^{(k)}$  as follows:

$$\mathcal{M}_{|\mathcal{V}_k|}^{(k)} = \{\mathbf{x}^{(k)} \in \mathbb{R}^{3|\mathcal{V}_k|} :$$

$$R_{\eta_k}(\mathbf{x}_{i+1}^{(k)} - \mathbf{x}_i^{(k)}) = R_{2\pi/n} R_{\eta_k}(\mathbf{x}_{i+2}^{(k)} - \mathbf{x}_{i+1}^{(k)}), \quad (14a)$$

$$i = 1, \dots, |\mathcal{V}_k| - 2,$$

$$\mathbf{e}_z^T R_{\eta_k}(\mathbf{x}_{|\mathcal{V}_k|}^{(k)} - \mathbf{x}_{|\mathcal{V}_k|-1}^{(k)}) = \mathbf{e}_z^T R_{\eta_k}(\mathbf{x}_1^{(k)} - \mathbf{x}_{|\mathcal{V}_k|}^{(k)})\}, \quad (14b)$$

where  $R_{\eta_k}$  is a rotation matrix defined such that  $R_{\eta_k}^T \mathbf{e}_z = \mathbf{n}_k$ . The desired Johnson polyhedral formation is the convex hull of a vertex set which can be represented as a subspace  $\mathcal{M}_n \subset \mathbb{R}^{3n}$  defined by the polygonal constraints that stem from each face in  $\mathcal{Q}$ , that is

$$\mathcal{M}_n = \bigcap_{k: Q_k \in \mathcal{Q}} \mathcal{M}_{|\mathcal{V}_k|}^{(k)}. \quad (15)$$

By Lemma III.3, the constraints in (14a) and (14b) can be compactly represented as the null space of a  $3(|\mathcal{V}_k| - 2) + 1 \times 3n$  matrix, denoted by  $\tilde{V}^{(k)}$ . In other words,  $\mathbf{x} \in \mathcal{M}_{|\mathcal{V}_k|}^{(k)} \Leftrightarrow \tilde{V}^{(k)} \mathbf{x} = \mathbf{0}$ . Using (11) and (13), one can write  $\tilde{V}^{(k)}$  as

$$\tilde{V}^{(k)} = \mathcal{W}_{|\mathcal{V}_k|} \mathcal{P}_{|\mathcal{V}_k|} \mathcal{R}_{\eta_k} E^{(k)}. \quad (16)$$

Thus, the subspace  $\mathcal{M}_n$  in (15) may be equivalently re-stated as the intersection of null spaces:

$$\mathcal{M}_n = \bigcap_{k: Q_k \in \mathcal{Q}} \mathcal{N}(\tilde{V}^{(k)}).$$

Note that the rules  $\mathcal{R}$  are implicitly encoded in the indexing used to write down the constraint matrices  $\tilde{V}^{(k)}$  and their nullspace intersection. Given our representation of the PPS in terms of the set  $\mathcal{Q}$ , we can now combine (15) and (16) to form the global constraint matrix  $\tilde{V}$ :

$$\tilde{V} := [\tilde{V}^{(1)^T} \quad \tilde{V}^{(2)^T} \quad \dots \quad \tilde{V}^{(L)^T}]^T. \quad (17)$$



The global formation subspace  $\mathcal{M}_n$  is then defined to be the null space of  $\tilde{V}$ , i.e.,

$$\mathbf{x} \in \mathcal{M}_n \Leftrightarrow \tilde{V}\mathbf{x} = \mathbf{0}.$$

As constructed,  $\tilde{V}$  is not yet full row rank. A geometric explanation for this linear dependence is that once the orientations of two adjacent PPS faces are fixed, the in-plane rotational degrees of freedom for the remaining faces vanish. This notion is formalized in the following (relatively straightforward) corollary of Lemma III.1, the proof of which is provided in the appendix.

**Corollary IV.5** (Reduced Polygon Constraints). *Assume two neighboring robots  $j$  and  $j+1$  of face  $\mathcal{Q}_k$  are constrained to lie in a plane normal to  $\mathbf{n}_k$ , where the indices  $\{j, j+1\} \in \{1, \dots, |\mathcal{V}_k|\}$  are modulo  $|\mathcal{V}_k|$ . Then the rotational constraints given in (14a) are necessary and sufficient for the definition of a regular polygon with normal direction  $\mathbf{n}_k$ .*

Since stability analysis using partial contraction is contingent upon a full row rank constraint matrix, we present a reduction of  $\tilde{V}$  that discards redundant constraints. Consider the following reduced constraint matrix  $V$  with similar structural properties as  $\tilde{V}$ :

$$V := \begin{bmatrix} V^{(1)T} & V^{(2)T} & \dots & V^{(L)T} \end{bmatrix}^T, \quad (18)$$

$$V^{(k)} = \begin{cases} \tilde{V}^{(k)} & \text{if } k = 1, 2, \\ (W_{r|\mathcal{V}_k|} \otimes I_3) \mathcal{P}_{|\mathcal{V}_k|} \mathcal{R}_{\eta_k} E^{(k)} & k = 3, \dots, L, \end{cases} \quad (19)$$

where, as defined in Lemma III.3,  $W_{r|\mathcal{V}_k|} = [I_{|\mathcal{V}_k|-2}, 0_{(|\mathcal{V}_k|-2) \times 2}]$ . That is, in-plane constraints have been removed from all but two (adjacent) faces within  $\mathcal{Q}$ , specifically,  $\mathcal{Q}_1$  and  $\mathcal{Q}_2$ . The following lemma shows that the reduced constraints that stem from (19) are equivalent to the original set of constraints using  $\tilde{V}^{(k)}$ , and that the resulting global constraint matrix  $V$  is full row rank.

**Lemma IV.6** (Minimal Representation of  $V$ ). *Denote the set of equations  $\tilde{V}^{(k)}\mathbf{x} = \mathbf{0}$  where  $\tilde{V}^{(k)}$  has the form given in (16) for  $k = 1, \dots, L$  as the full-constraint set. Similarly, denote the set of equations  $V^{(k)}\mathbf{x} = \mathbf{0}$  where  $V^{(k)}$  has the form given in (19) as the reduced-constraint set. Then, the solutions to the two sets of equations are identical. That is,*

$$\mathbf{x} \in \mathcal{M}_n \Leftrightarrow V\mathbf{x} = \mathbf{0}.$$

Furthermore,  $V$  is full row rank.

*Proof.* We assume without loss of generality that the faces  $\mathcal{Q}_k$  (and thus the corresponding constraint matrices  $V^{(k)}$ ) are ordered so that for any  $j = 1, \dots, L$  all faces  $\mathcal{Q}_i$  along the unique path between  $\mathcal{Q}_1$  and  $\mathcal{Q}_j$  satisfy  $i \leq j$ . Let  $\mathcal{Q}_k = \{\mathcal{Q}_1, \dots, \mathcal{Q}_k\}$  denote a partial list of faces, constrained by the set of equations  $[V^{(k)}]\mathbf{x} = \mathbf{0}$ , where  $[V^{(k)}] = \begin{bmatrix} V^{(1)T} & \dots & V^{(k)T} \end{bmatrix}^T$ ,  $k \leq L$ . Similarly, let  $[\tilde{V}^{(k)}] = \begin{bmatrix} \tilde{V}^{(1)T} & \dots & \tilde{V}^{(k)T} \end{bmatrix}^T$ . We proceed by induction.

Base case  $k = 2$ : It is clear that  $\mathcal{N}([V^{(2)}]) = \mathcal{N}([\tilde{V}^{(2)}])$  as  $[V^{(2)}] = [\tilde{V}^{(2)}]$ . To see that  $[V^{(2)}]$  is full row rank, we note that  $V^{(1)}$  is full row rank (Lemma III.1) and each successive rotational constraint within  $V^{(2)}$  (counting around the polygon starting from the shared edge) involves at least one robot that

does not belong to any previous constraints in  $V^{(1)}$  or  $V^{(2)}$ . The in-plane constraint for  $V^{(2)}$  further reduces the dimension of the solution set by constraining the rotational degree of freedom for  $\mathcal{Q}_1$ : there is only one orientation of  $\mathcal{Q}_1$  (up to reflection, i.e. negative scaling) under which the edge shared by the two faces is orthogonal to  $\mathbf{n}_2$ . Therefore all constraints in  $[V^{(2)}]$  are independent.

**Inductive Hypothesis:** Assume that the set of equations  $[\tilde{V}^{(k)}]\mathbf{x} = \mathbf{0}$  and  $[V^{(k)}]\mathbf{x} = \mathbf{0}$  are equivalent and that  $[V^{(k)}]$  is full row rank.

We now prove that  $[\tilde{V}^{(k+1)}]\mathbf{x} = \mathbf{0}$  and  $[V^{(k+1)}]\mathbf{x} = \mathbf{0}$  are also equivalent and that  $[V^{(k+1)}]$  is full row rank. The inductive hypothesis  $[V^{(k)}]\mathbf{x} = \mathbf{0}$  encodes rules which assemble the faces of  $\mathcal{Q}_k$  (which form a tree by the ordering assumption) into a unique PPS. Now, from our ordering assumption,  $\mathcal{Q}_{k+1}$  must share exactly one edge (i.e., exactly 2 robots) with a face within  $\mathcal{Q}_k$ , and that the edge must lie in a plane normal to  $\mathbf{n}_{k+1}$ . The tree structure of  $\mathcal{Q}_k$  prevents any additional shared edges. From Corollary IV.5, the rotational constraints encoded in  $V^{(k+1)}$  are indeed sufficient to fully constrain  $\mathcal{Q}_{k+1}$ . Thus,  $[V^{(k+1)}]\mathbf{x} = \mathbf{0}$  and  $[\tilde{V}^{(k+1)}]\mathbf{x} = \mathbf{0}$  are equivalent.

To see that  $[V^{(k+1)}]$  is full row rank, we note that each successive rotational constraint within  $V^{(k+1)}$  involves at least one robot not represented by any of the existing constraints within  $[V^{(k)}]$  or  $V^{(k+1)}$ . Then, by similar reasoning as for the base case, all constraints in  $[V^{(k+1)}]$  are independent.

By induction, both claims are proven.  $\square$

**Remark IV.7** (Polyhedron Degrees of Freedom). *Note that  $V$  has a total of  $\left(3 \sum_{k=1}^L |\mathcal{V}_k| - 6L + 2\right)$  linearly independent equations for a total of  $\left(3 \sum_{k=1}^L |\mathcal{V}_k| - 6L + 6\right)$  variables. Three degrees of freedom correspond to the three translational degrees of freedom. The final degree of freedom corresponds to scaling the polyhedron by some constant  $\alpha \in \mathbb{R}$ . Applying a negative scaling factor may be interpreted as inverting the entire formation.*

## B. Polyhedral Formation Control Law

Having characterized the desired formation as the null space of a full row rank matrix, we turn our attention to the control laws. The control law for each robot is given by the sum of a number of contributions, one for each face in which the robot represents a vertex. Specifically, let  $\mathbf{u}_i^{(k)}$  denote the contribution to the control law for robot  $i$  due to face  $k$ , obtained by generalizing the symmetric cyclic controller presented in Section III-C, namely:

$$\mathbf{u}_i^{(k)} = \sum_{m=1}^{N_k} k_m^{(k)} [R_{m_s}^{(k)} (\mathbf{x}_{i+m}^{(k)} - \mathbf{x}_i^{(k)}) + R_{m_s}^{(k)T} (\mathbf{x}_{i-m}^{(k)} - \mathbf{x}_i^{(k)})], \quad (20)$$

where  $N_k < |\mathcal{V}_k| - 1$  is the look-ahead horizon for face  $k$ ,  $k_m^{(k)} > 0$  is a gain,  $R_{m_s}^{(k)} = R_{\eta_k}^T R_m^{(k)} R_{\eta_k}$  with  $R_m^{(k)} = R_{m\pi/|\mathcal{V}_k|}$ , and  $\mathbf{x}_{i+m}^{(k)}$  and  $\mathbf{x}_{i-m}^{(k)}$  are neighboring robots within face  $k$  (modulo  $|\mathcal{V}_k|$ ). The choice  $R_m^{(k)} = R_{m\pi/|\mathcal{V}_k|}$  stems from the requirement that robots converge to a polygon of fixed size within each face (see Remark III.6).

The net control for each robot is then given by the superposition of the contributions from each face in which the robot is a vertex, i.e.,

$$\mathbf{u}_i = \sum_{k:i_k \neq \infty}^L \mathbf{u}_i^{(k)}, \quad i = 1, \dots, n. \quad (21)$$

The control laws in (20) and (21) highlight the information requirements for each robot in the formation. In particular, we assume that each robot knows: (a) what face(s) it belongs to, (b) relative position with respect to its forward and rear neighbors within each face, (c) the number of robots in these faces, and (d) the rotation matrix  $R_\eta$  that accompanies the relative position measurements. In a scenario where there are a large number of robots in the overall formation, these information requirements are fairly minimal, emphasizing the decentralized framework that underpins this work.

Using (20) and (10), the overall control vector stemming from face  $k$  is given by

$$\mathbf{u}^{(k)} = - \sum_{m=1}^{N_k} k_m^{(k)} \mathcal{L}_{m\eta}^{(k)} \mathbf{x}^{(k)} := -\mathcal{L}_\eta^{(k)} E^{(k)} \mathbf{x},$$

where  $\mathcal{L}_{m\eta}^{(k)} = \mathcal{R}_{\eta k}^T \mathcal{L}_m^{(k)} \mathcal{R}_{\eta k}$  and  $\mathcal{L}_m^{(k)} = L_m \otimes R_m^{(k)} + L_m^T \otimes R_m^{(k)T}$ . The overall closed-loop dynamics are then given by

$$\dot{\mathbf{x}} = \mathbf{u} = \sum_{k=1}^L -E^{(k)T} \mathcal{L}_\eta^{(k)} E^{(k)} \mathbf{x}. \quad (22)$$

For simplicity, we are considering zero internal dynamics. A sufficient condition for convergence to the desired formation, i.e., to the subspace  $\mathcal{M}_n$ , is provided by the following theorem.

**Theorem IV.8** (Polyhedron Convergence). *Assume*

$$J = \left[ V \left( \sum_{k=1}^L -E^{(k)T} \mathcal{L}_\eta^{(k)} E^{(k)} \right) V^T \right] \prec 0. \quad (23)$$

*Then, the closed-loop dynamics (22) globally converge to a Johnson polyhedral formation, i.e., to the formation subspace  $\mathcal{M}_n$ .*

*Proof.* Let  $\bar{V}$  represent the orthonormal counterpart of  $V$ , whose existence is guaranteed given the results of Lemma IV.6. First, we need to show that the dynamics in (22) are flow invariant. Indeed, as an immediate consequence of Remark III.6 and the fact that the rotation angle is set equal to  $m\pi/|\mathcal{V}_k|$ , one has

$$\mathbf{x} \in \mathcal{M}_n \Rightarrow \mathbf{u}^{(k)}(\mathbf{x}) = \mathbf{0}, \quad \text{for all } k = 1, \dots, L, \quad (24)$$

and hence  $\mathbf{x} \in \mathcal{M}_n \Rightarrow \bar{V} \mathbf{u} = \mathbf{0}$ , i.e., the dynamics in (22) are flow invariant.

Consider, then, the following auxiliary system for (22):

$$\dot{\mathbf{y}} = \bar{V} \left( \sum_{k=1}^L -E^{(k)T} \mathcal{L}_\eta^{(k)} E^{(k)} (\bar{V}^T \mathbf{y} + \bar{U}^T \bar{U} \mathbf{x}_p) \right).$$

Since the dynamics are flow invariant, and an invertible transformation exists between  $V$  and  $\bar{V}$ , then by Remark II.4 and by applying Corollary II.3, one obtains the claim.  $\square$

**Remark IV.9** (Conditions for Polyhedron Convergence). *Note that the condition given in (23) is markedly more complex than*

*(9), required for convergence to a single regular polygon. In particular, we now have the addition of cross-terms in the projected Jacobian due to the interactions between the various faces. Additionally, when working with the auxiliary system, we did not leverage the results of flow-invariant subspaces as they were introduced in Section III. This is because if the global state vector  $\mathbf{x}$  converges to one of the subspaces defined by  $V^{(k)}$ , then the global dynamics will not necessarily be flow-invariant with respect to the given face or any other faces in  $\mathcal{M}_n$ . This is due to two reasons: (a) only the rotational set of constraints are used to define faces other than the ones indexed by  $k = 1$  and  $k = 2$ , which we know are insufficient on their own to describe a regular planar polygon, and (b) the symmetric cyclic control law for each robot as defined in (21) introduces coupling between various faces. Thus, convergence is predicated on the decay of the symmetric cyclic controller to zero for all faces.*

Figure 4 shows simulation results demonstrating convergence to example 3D formations.

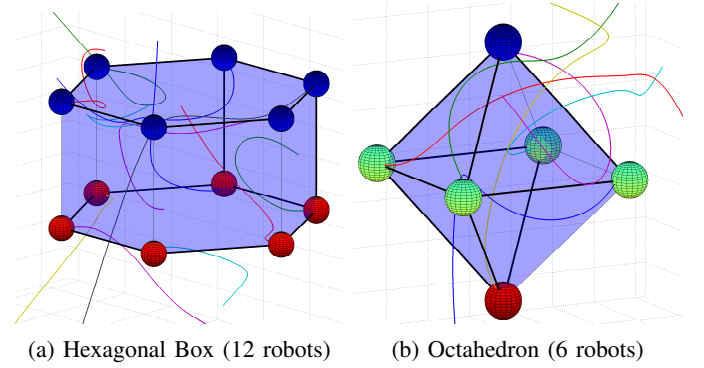
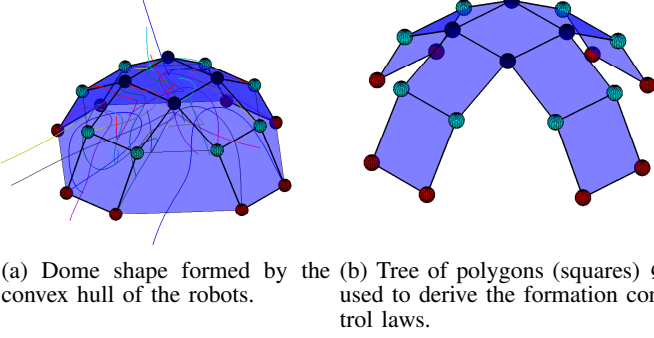


Fig. 4: Convergence to polyhedra. Faces with bold edges correspond to the set  $\mathcal{Q}$ .

**Remark IV.10** (Nonsymmetric Tree-Based Formations). Lemma IV.4 sets forth the essential properties of polyhedral surface developments required for the preceding control law and convergence analysis to apply. That is, viable targets under this theory consist of any formation of robots expressible as the vertices of spatially-oriented regular polygons arranged according to a tree structure. Johnson solid formations are a particularly motivating example, given their symmetry, but other grid and mesh formations are also possible. For instance, Figure 5 illustrates an example where a curved dome is represented by the vertices of a tree of squares ( $\mathcal{Q}$ ), and the symmetric cyclic controller is defined and assembled using equations (20) and (21). Note that although there are 20 robots in this formation, each robot only requires relative position measurements with respect to three other robots (the robots in the top square require relative measurements with respect to four other robots) in the group. Thus, in addition to highlighting the applicability of our approach to the formation control of nonsymmetric formations, this example also underlines the sparsity of the decentralized formation control laws.

In future work, we intend to investigate strategies for accommodating local robot failures. For instance, in the regular polygon case, a single point of failure simply requires the robots to reinitialize their indices within the group and subsequently converge to a smaller polygon. In the polyhedron case this can be problematic if the failed robot is coupled





(a) Dome shape formed by the convex hull of the robots. (b) Tree of polygons (squares)  $Q$  used to derive the formation control laws.

Fig. 5: Dome-shaped formation with 20 robots. The formation may be extended to collectively film sports or entertainment events.

to multiple faces. In this scenario, a global restart may be required with a new configuration that fits the reduced set of robots.

## V. EXTENSIONS

In this section we present a number of extensions to the symmetric cyclic controller discussed in Sections III and IV. In Subsection V-A, we provide extensions to control the formation size, and investigate convergence properties under delays. Subsection V-B details a heuristic control law for preventing collisions through use of Repulsive Potential Functions. Finally, in Subsection V-C, we study the disturbance rejection properties of the symmetric cyclic controller given in (7) with respect to bounded additive disturbances.

### A. Control Over Formation Size

We first address control over the size of a regular polygon formation. By setting  $\alpha_m = m\pi/n$  in (6), the formation converges to a regular polygon of fixed, but uncontrolled size (in particular, the size depends on the initial conditions). In [16], the authors introduce an additional term to (6) to control the size of the formation, however, the auxiliary system used to study convergence using partial contraction does not admit the necessary stationary zero solution. We present a different approach to the size control problem by simply modifying the rotation angle  $\alpha_m$  embedded within the cyclic controller.

Let  $\rho$  be the desired polygon side-length for the formation and define  $p_i(\mathbf{x}) := 1 - \|\mathbf{x}_{i+1} - \mathbf{x}_i\|/\rho$  to be the normalized error in the inter-robot distance between robots  $i$  and  $i+1$ , where  $\|\cdot\|$  denotes the Euclidean norm. Let  $\bar{p}(\mathbf{x})$  be the arithmetic average over all  $p_i(\mathbf{x})$  (hereby referred to as the average inter-robot distance error), and define  $f_s(\bar{p})$  to be a continuous odd scalar function that satisfies the properties: (a)  $\bar{p}f_s(\bar{p}) > 0$  for  $\bar{p} \neq 0$ , (b)  $|f_s(\bar{p})| \leq 1$ , and (c)  $\exists a \in \mathbb{R}_{>0}$  such that  $f'_s(0) > a$ . The last (technical) property is required to ensure sufficient control authority near  $\bar{p} = 0$  and will be formalized in Theorem V.2. Some examples for  $f_s$  include the hyperbolic tangent function, or the saturation function ( $|f_s(p_i)| = \min\{|p_i|, 1\}$ ). Consider the following modification

to the symmetric cyclic control law introduced in (6), now expressed as a *hybrid* dynamical system:

$$\mathbf{u}_i(t) = \sum_{m=1}^N k_m [R_m(\mathbf{x})(\mathbf{x}_{i+m} - \mathbf{x}_i) + R_m(\mathbf{x})^T(\mathbf{x}_{i-m} - \mathbf{x}_i)],$$

$$t \in [k\tau, (k+1)\tau), \quad k \in \mathbb{N}, \quad (25)$$

where  $R_m(\mathbf{x})$  is a rotation matrix around  $\mathbf{e}_z$  with rotation angle  $m\pi/n + \alpha_s(\mathbf{x})$ , where  $\alpha_s(\mathbf{x}) := \alpha_{s0} f_s(\bar{p}(\mathbf{x}((k-1)\tau)))$ ,  $\alpha_{s0} \in \mathbb{R}_{>0}$  is a gain, and  $\tau \in \mathbb{R}_{>0}$  is a constant time lag.

In equation (25) the rotation angle has been modified *equally* for all robots by the amount  $\alpha_s$ , which is a function of the average inter-robot distance error  $\bar{p}$ . The time lag parameter  $\tau$  accounts for the finite time required to compute  $\bar{p}$ , e.g., by using discrete average consensus techniques [26]. That is, the angle  $\alpha_s$  used in the time interval  $[k\tau, (k+1)\tau)$  is computed using the average inter-robot distance error at time  $(k-1)\tau$ . To simplify notation, we will use  $(\mathbf{x}, \tau)$  to reference the continuous/discrete hybrid dynamics expressed in (25).

As the rotation angle modification is the same for all robots, the net control law can still be expressed using circulant matrices:

$$\dot{\mathbf{x}} = \mathbf{u} = - \underbrace{\sum_{m=1}^N k_m [L_m \otimes R_m(\mathbf{x}, \tau) + L_m^T \otimes R_m(\mathbf{x}, \tau)^T]}_{:=\mathcal{L}(\mathbf{x}, \tau)} \mathbf{x}. \quad (26)$$

To analyze convergence to the desired formation, first define  $\mathcal{M}_{n\rho}$  to be a subset of the original invariant subspace  $\mathcal{M}_n$ , corresponding to the space of regular polygon formations with inter-robot neighbor distance equal to  $\rho$ . Thus,

$$\mathcal{M}_{n\rho} = \{\mathbf{x} : \mathbf{x} \in \mathcal{M}_n \wedge \|\mathbf{x}_{i+1} - \mathbf{x}_i\| = \rho, \forall i\}. \quad (27)$$

To prove convergence to this subspace, we first start with a lemma to prove that the subspace  $\mathcal{M}_n$  is indeed flow-invariant with respect to (26).

**Lemma V.1** (Flow-invariance). *The subspace  $\mathcal{M}_n$  is flow-invariant with respect to the dynamics given in (26).*

*Proof.* Given that the symmetric cyclic controller can still be expressed using circulant matrices as in (26), by Lemma III.4, the claim follows.  $\square$

Having shown that  $\mathcal{M}_n$  (which contains  $\mathcal{M}_{n\rho}$ ) is flow-invariant with respect to our modified dynamics, we now present sufficient conditions to ensure that  $\mathbf{x}$  converges to the manifold  $\mathcal{M}_{n\rho}$ . The proof for the following theorem relies on several intermediate results which are detailed in the appendix.

**Theorem V.2** (Polygon Convergence with Desired Size). *Let  $C = 2\Gamma$ , where*

$$\Gamma = \begin{cases} \frac{\sqrt{2(1 - \cos(2\pi/n))}}{\sin(\frac{\pi}{n})} \sum_{m=1}^N k_m & \text{if } n \text{ is even,} \\ \frac{\sqrt{2(1 - \cos(2\pi/n))}}{2 \sin(\frac{\pi}{2n})} \sum_{m=1}^N k_m & \text{if } n \text{ is odd.} \end{cases}$$

*Assume*

$$\inf_{\alpha_m \in [m\pi/n - \alpha_{s0}, m\pi/n + \alpha_{s0}]} \left( \min_{\substack{1 \leq i \leq n \\ k \in \{-1, 0, 1\}}} \sum_{m=1}^N k_m \lambda_{ik}^{(m)} \right) > 0, \quad (28)$$

where  $\lambda_{ik}^{(m)}$  has the form given in Theorem III.5, and suppose the time lag parameter  $\tau$  satisfies the following bound:

$$\tau < \min \left\{ \frac{1}{C}, \frac{1}{8CT} \right\}, \quad (29)$$

where  $T$  denotes a positive constant such that  $(1/2)T|\bar{p}| \leq |\sin(\alpha_{s_0} f_s(\bar{p}))| \leq T|\bar{p}|$  for  $|\bar{p}| < 1$  (if  $f_s$  is the saturation function,  $T = \alpha_{s_0}$ ).

Then the robots converge to a regular polygon with the desired side-length  $\rho$ . That is,  $\mathbf{x}$  converges to the manifold  $\mathcal{M}_{n\rho}$ .

**Remark V.3** (Time Lag Bound). Note that the bound for the time lag parameter  $\tau$  given in (29) can be relaxed by using a value for  $\Gamma$  that is strictly smaller than the one given in Theorem V.2. The derivation of this relaxed bound is detailed in the appendix.

**Remark V.4** (Control Over Formation Center). The control laws may be trivially extended to also permit control over the geometric center of the formation. For instance, let  $\mathbf{x}_c$  represent the desired center and let  $\mathbf{x}_0 := (1/n) \sum_{i=1}^n \mathbf{x}_i$  be the instantaneous geometric center. In similar spirit to the continuous/discrete dynamics in (25), the center control law  $\mathbf{u}_c$  for robot  $i$  may be expressed as:

$$\mathbf{u}_c(t) = k_c (\mathbf{x}_c - \mathbf{x}_0((k-1)\tau)), t \in [k\tau, (k+1)\tau), k \in \mathbb{N},$$

where  $k_c \in \mathbb{R}_{>0}$  is a gain and the time lag parameter  $\tau$  accounts for the finite time required to compute  $\mathbf{x}_0$  in a decentralized fashion (e.g., using message passing). By leveraging analysis similar to that presented in the proof for Theorem V.2, it is possible to derive a bound on the steady state error  $\|\mathbf{x}_c - \mathbf{x}_0\|$ .

**Remark V.5** (Size Control Extension to Polyhedra). The size controller discussed in this subsection may be trivially extended to the formation control of polyhedra by decoupling the polygon  $\mathcal{Q}_1$  from all other polygons  $\mathcal{Q}$  in the PPS. The net control for all robots in  $\mathcal{Q}_1$  is given by the modified symmetric cyclic controller in (26) plus a “rotational controller” that rotates the robots within the plane such that the first face has an orientation compatible with adjacent faces within the PPS. The control for all other robots in the polyhedron is given by the standard symmetric cyclic controller as given in (20) and (21). The proof for convergence is then also decoupled by treating  $\mathcal{Q}_1$  as an independent polygon with respect to the remaining PPS. An example rotational controller along with a proof of correctness is provided in the appendix.

### B. Collision Avoidance

For collision avoidance, one must account for the size of each robot in the formation. In particular, we assume that each robot is completely contained within a sphere of radius  $r_1$ . To design a collision avoidance controller, we make use of Repulsive-Potential-Functions (RPF) [27].

Consider a “collision-detection” zone around each robot parametrized by two variables  $r_1 < r_2$ , where  $r_2$  denotes the radius of the boundary of a robot’s detection region. The RPF between any two robots  $i$  and  $j$  can be written as follows:

$$V_{ij}(d_{ij}) := -\frac{(r_2 - r_1)^2}{r_1 - d_{ij}} + 2(r_2 - r_1) \log(r_1 - d_{ij}) - d_{ij} - v(r_1, r_2), \text{ if } r_1 < d_{ij} \leq r_2, \quad (30)$$

and equals zero for  $d_{ij} > r_2$ , where  $d_{ij} = \|\mathbf{x}_j - \mathbf{x}_i\|$  and  $v(r_1, r_2) := 2(r_2 - r_1) \log(r_1 - r_2) - r_1$  is a constant. The normalized force magnitude can be derived from the RPF by taking the derivative with respect to  $d_{ij}$  giving:

$$f_{c_{ij}}(d_{ij}) = \begin{cases} \text{undefined} & \text{if } d_{ij} \leq r_1, \\ \frac{(d_{ij} - r_2)^2}{d_{ij}(d_{ij} - r_1)^2} & \text{if } r_1 < d_{ij} \leq r_2, \\ 0 & \text{if } d_{ij} > r_2. \end{cases} \quad (31)$$

Notice that both  $V_{ij}$  and  $f_{c_{ij}}$  asymptotically approach  $+\infty$  as  $d_{ij} \rightarrow r_1$ , and are  $\mathcal{C}^1$  continuous for  $d_{ij} > r_1$ . We now formulate the collision avoidance controller for robot  $i$  as follows:

$$\mathbf{u}_{\text{coll}_i} = - \sum_{j \neq i}^n f_{c_{ij}}(d_{ij})(\mathbf{x}_j - \mathbf{x}_i). \quad (32)$$

Thus the collision avoidance controller adds a non-zero (flow-invariant with respect to  $\mathcal{M}_n$ ) “escape” velocity to the existing symmetric cyclic controller for robot  $i$  if and only if there exists another robot  $j$  within the  $i^{\text{th}}$  robot’s collision avoidance zone.

Notice that  $f_{c_{ij}}$  is unbounded as  $d_{ij} \rightarrow r_1$  which makes convergence analysis with respect to the subspace  $\mathcal{M}_n$  intractable. In particular, it becomes difficult to upper bound the maximum eigenvalue of the Jacobian of the collision avoidance controller. In lieu of this analysis, we present Monte Carlo simulation results in Section VI for six quadcopters attempting to converge to a polygon while implementing the collision avoidance controller above.

### C. Robustness Analysis

To conclude our analysis of the symmetric cyclic control algorithm, we investigate the robustness properties of the closed-loop dynamics given in (7) in the presence of additive disturbances. We quantify robust performance by the Euclidean norm of the deviation of the perturbed system trajectory from the nominal contracting trajectory with respect to the metric  $\bar{V}^T \bar{V}$ , where  $\bar{V}$  is the orthonormal counterpart of the nominal projection matrix as defined in (5). The analysis presented in this section is restricted to zero internal dynamics (i.e.,  $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ ) and to convergence to regular polygons. An analogous proof can be derived for the Johnson polyhedron case.

Let  $\mathbf{z}(t) = \bar{V}\mathbf{x}(t)$  represent the dynamics of the unperturbed system under the action of the symmetric cyclic controller (7). Then,

$$\dot{\mathbf{z}}(t) = -\bar{V}\mathcal{L}\mathbf{x}(t). \quad (33)$$

Consider the perturbed dynamics  $\mathbf{z}_d(t)$  under an additive state and/or time-dependant disturbance:

$$\dot{\mathbf{z}}_d(t) = -\bar{V}\mathcal{L}\mathbf{x}_d(t) + \mathbf{d}(\mathbf{x}_d, t). \quad (34)$$

Here the disturbance is measured in the transformed  $\mathbf{z}$  coordinates. Let  $\delta(t)$  represent the difference between the nominal and perturbed trajectories at time  $t$ . Specifically, let  $\delta_{\mathbf{z}}(t) := \mathbf{z}_d(t) - \mathbf{z}(t)$  and  $\delta_{\mathbf{x}}(t) := \mathbf{x}_d(t) - \mathbf{x}(t)$ . Note that both  $\delta_{\mathbf{z}}$  and  $\delta_{\mathbf{x}}$  equal  $\mathbf{0}$  at  $t = 0$  as the trajectories are assumed to start from the same initial conditions. Then,

$$\dot{\delta}_{\mathbf{z}}(t) = -\bar{V}\mathcal{L}\delta_{\mathbf{x}}(t) + \mathbf{d}(\mathbf{x}_d, t). \quad (35)$$

Let  $\bar{R}^2(t) := \delta_z^T(t)\delta_z(t)$ ;  $\bar{R}$  represents the Euclidean distance of the perturbed trajectory from the nominal one in the transformed coordinates, referred to as the *formation error*. The following theorem characterizes the robustness of the symmetric cyclic controller.

**Theorem V.6** (Cyclic Control Robustness). *Assume that the nominal system is contracting with contraction rate  $\Lambda = -\lambda_{\min}(\bar{V}\mathcal{L}\bar{V}^T) < 0$  and that the disturbance is norm bounded with upper bound  $\bar{d}$ . Then the formation error in the transformed  $z$  coordinates, that is  $\bar{R}$ , is upper bounded as*

$$\bar{R}(t) \leq \frac{\bar{d}}{\Lambda}(e^{\Lambda t} - 1). \quad (36)$$

*Proof.* Starting from (35),  $\delta_x = (\bar{V}^T\bar{V} + \bar{U}^T\bar{U})\delta_x = \bar{V}^T\delta_z + \bar{U}^T\bar{U}\delta_x$ . Now,  $\bar{U}^T\bar{U}\delta_x \in \mathcal{M}_n$ , so  $\bar{V}\mathcal{L}(\bar{U}^T\bar{U}\delta_x) = \mathbf{0}$ . Thus,

$$\dot{\delta}_z(t) = -\bar{V}\mathcal{L}\bar{V}^T\delta_z + \mathbf{d}(x_d, t).$$

Pre-multiplying by  $2\delta_z(t)$  we obtain:

$$2\delta_z^T\dot{\delta}_z = \frac{d(\delta_z^T\delta_z)}{dt} = -2\delta_z^T(\bar{V}\mathcal{L}\bar{V}^T)\delta_z + 2\delta_z^T\mathbf{d}(x_d, t).$$

Letting  $\bar{R}^2 = \delta_z^T\delta_z$ , one can write

$$\frac{d\bar{R}^2}{dt} = 2\bar{R}\frac{d\bar{R}}{dt} = -2\delta_z^T(\bar{V}\mathcal{L}\bar{V}^T)\delta_z + 2\delta_z^T\mathbf{d}(x_d, t).$$

Noting that:

$$-\delta_z^T(\bar{V}\mathcal{L}\bar{V}^T)\delta_z \leq -\lambda_{\min}(\bar{V}\mathcal{L}\bar{V}^T)\delta_z^T\delta_z = \Lambda\bar{R}^2,$$

and

$$\delta_z^T\mathbf{d}(x_d, t) \leq \|\delta_z\|\|\mathbf{d}(x_d, t)\| = \bar{R}\|\mathbf{d}(x_d, t)\|,$$

we get:

$$\bar{R}\dot{\bar{R}} \leq \Lambda\bar{R}^2 + \bar{R}\|\mathbf{d}(x_d, t)\|.$$

Thus,

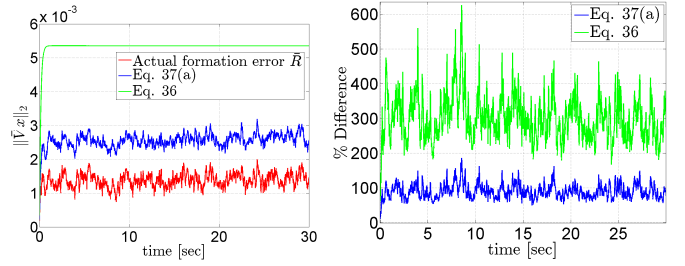
$$\dot{\bar{R}} \leq \Lambda\bar{R} + \|\mathbf{d}(x_d, t)\|, \quad (37a)$$

$$\leq \Lambda\bar{R} + \bar{d} \quad (37b)$$

By the comparison theorem [28], the claim follows.  $\square$

**Remark V.7.** *The bound  $\bar{R}(t)$  describes the time-varying deviation from a nominal contracting trajectory. In the limit  $t \rightarrow \infty$ , the steady state bound  $\bar{R}_{ss} := \bar{d}/|\Lambda|$  provides a measure of the steady state formation error. Physically, it represents a bound on the degree of asymmetry at each node of the formation, as measured by the set of linear constraints used to define the subspace, that is, (4a) and (4b). The inverse proportionality between  $\bar{R}_{ss}$  and  $|\Lambda|$  highlights a trade-off between increased robustness, control saturation, and sensor requirements.*

Figure 6 provides insights into the tightness of the bound. Here, disturbances were introduced into the system via error in the control rotation angles. The rotation angle in  $R_m$  for robot  $i$  is equal to  $\hat{\alpha}_i(t) = \alpha_0 + \delta_{\alpha_i}(t)$  where  $\alpha_0 = m\pi/n$  is the nominal control angle and  $\delta_{\alpha_i}(t)$  is a perturbation, sampled randomly from a uniform distribution over the range  $[\delta_{\alpha_i}, \bar{\delta}_{\alpha_i}]$  where  $\delta_{\alpha_i}, \bar{\delta}_{\alpha_i} \in [-1, 1]^\circ$ . The plotted curves in Figure 6a correspond to the actual formation error  $\bar{R}$ , the bound in (36), and the numeric integration of the expression in (37a). Figure 6b quantifies the tightness of (36) and (37a) by plotting the percentage difference with respect to the actual deviation.



(a) Comparison for  $\|\delta_z\|$ .

(b) Bound tightness.

Fig. 6: Tightness for robustness bounds. Parameters:  $n = 6$ ,  $N = 2$ ,  $\Lambda = -6.928$ ,  $\bar{d} = 0.065$ .

## VI. FORMATION CONTROL OF QUADCOPTERS

In this section we demonstrate the use of the symmetric cyclic control laws developed in Sections III, IV and V-A on a fleet of quadcopters.

### A. Quadcopter Dynamics and Control

Let  $\mathbf{x}_i$  and  $\mathbf{v}_i$  denote the position and inertial velocity respectively of the  $i^{th}$  quadcopter, resolved in an inertial frame with a downward pointing  $Z$  axis. Denote  $\mathbf{q}_i$  to be the attitude quaternion of the  $i^{th}$  quadcopter with associated Euler angles: Yaw ( $\psi_i$ ), Pitch ( $\theta_i$ ), and Roll ( $\phi_i$ ) with rotation order  $ZYX$ . The angular velocity of the  $i^{th}$  quadcopter with respect to the inertial frame, resolved in the quadcopter body frame, is given by  $\boldsymbol{\omega}_i$ . Let  $R_{ib}^i$  denote the rotation matrix to transform vectors in the inertial frame to the  $i^{th}$  quadcopter body frame. Finally let  $m_{q_i}$  and  $I_i$  denote the mass and moment of inertia tensor, respectively, for the  $i^{th}$  quadcopter.

The translational dynamics of the quadcopter are given by:

$$\begin{aligned} \dot{\mathbf{x}}_i &= \mathbf{v}_i, \\ \dot{\mathbf{v}}_i &= R_{ib}^{iT} \begin{bmatrix} 0 \\ 0 \\ -T_i/m_{q_i} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \end{aligned} \quad (38)$$

where  $T_i$  is the net thrust produced by all four propellers and  $g$  is the gravitational acceleration. Note that the direction of positive thrust is opposite the body  $z$ -axis. The rotational dynamics of the quadcopter are given by:

$$\begin{aligned} \dot{\mathbf{q}}_i &= \frac{1}{2} \begin{bmatrix} 0 \\ \boldsymbol{\omega}_i \end{bmatrix} \odot \mathbf{q}_i, \\ \dot{\boldsymbol{\omega}}_i &= I_i^{-1} \left( \mathbf{M}_i - \boldsymbol{\omega}_i \times I_i \boldsymbol{\omega}_i \right), \end{aligned} \quad (39)$$

where  $\odot$  denotes the quaternion product and  $\mathbf{M}_i$  is the net torque on the quadrotor as a result of differential thrust forces and reaction moments generated by the motor/propeller pairs. As the dynamics of the motors are significantly faster than those of the quadcopter as a rigid body, we assume that the net thrust force and moment generated by the propellers can be commanded instantaneously.

Notice that the rotation matrix  $R_{ib}^i$  couples the attitude and translational dynamics, thereby resulting in a fairly complex nonlinear dynamical system. In order to incorporate the symmetric cyclic control laws for formation control of quadcopters, we employ a two-level control hierarchy where, at the high level, the symmetric cyclic control laws in (6), (21) or (25) are used to generate a *desired* inertial velocity

reference for each quadcopter. The lower level control system then simply tracks this reference, and incorporates both thrust and attitude control. Several velocity tracking controllers for quadcopters have already been presented in the literature [29], [30]. In the appendix we present an example controller used for numerical simulations.

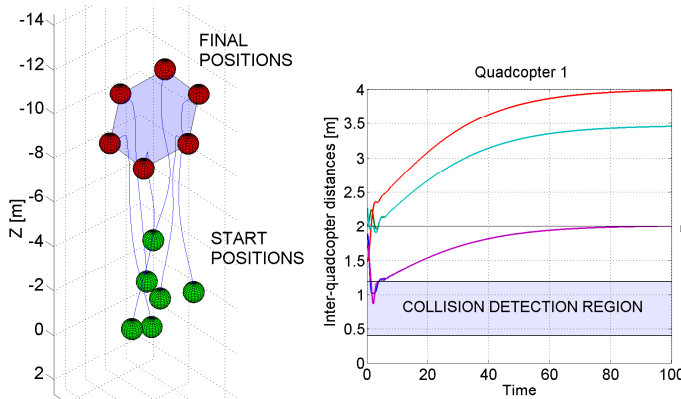
### B. Numerical Simulations and Experiments

We now present simulation and experimental results demonstrating the application of the above two-level control hierarchy on a group of six quadcopters. The quadcopters were initialized randomly in hover mode ( $\mathbf{v}_i = \mathbf{0}$ ) around the point  $[0, 0, -2]^T$ . The size control parameters were: desired inter-quadcopter distance  $\rho = 2.0m$ , function  $f_s(\bar{p}) = \tanh(\bar{p})$ , and gain  $\alpha_{s_0} = 5\pi/180$ . The desired formation center  $\mathbf{x}_c$  was selected as the point  $[0, 0, -10]^T m$ . The symmetric cyclic controller used a look-ahead horizon  $N = 1$  and gain  $k_1 = 0.5$ . The desired plane of convergence was tilted  $42^\circ$  with respect to the horizontal. We assumed a time lag  $\tau = 0.1s$ . The collision avoidance parameters were  $r_1 = 0.4m$ ,  $r_2 = 1.2m$ , and the controller was slightly modified to take advantage of relative velocity measurements:

$$\mathbf{u}_{\text{coll}_i} = - \sum_{j \neq i}^n \frac{(d_{ij} - r_2)^2}{d_{ij}(d_{ij} - r_1)^2} (\mathbf{x}_j - \mathbf{x}_i), \quad (40)$$

if  $r_1 < d_{ij} \leq r_2$  and  $v_{s_{ij}} < 0$ ,

where  $v_{s_{ij}} = (\mathbf{v}_j - \mathbf{v}_i) \cdot (\mathbf{x}_j - \mathbf{x}_i)/d_{ij}$  is the line-of-sight velocity between robots  $i$  and  $j$ . Additionally, the desired velocity (given by the sum of the cyclic, center and collision avoidance controllers) was subject to saturation constraints to prevent aggressive maneuvering and increased collision risk. In particular, each quadcopter was limited to a speed of  $3m/s$ . Figure 7a shows the resulting 3D trajectories as well as a shaded plane that outlines the final positions of the quadcopters. Figure 7b shows the evolution of the inter-quadcopter distances with respect to quadcopter 1. Notice how two neighboring quadcopters entered the collision detection region for quadcopter 1 before converging to the desired separation.



(a) Quadcopter 3D trajectories (b) Inter-quadcopter distances (in metres) for Quadcopter 1.

Fig. 7: Planar formation simulation.

To test the collision avoidance controller, we ran a Monte Carlo simulation with 100 samples randomizing the initial

position of the quadcopters within a ball of radius  $5m$  centered at  $\mathbf{x}_c$  (all simulations initialized the quadcopters in hover mode). Additionally, upon initialization, the quadcopters were assigned indices such that their projection onto the desired plane of convergence formed a non-intersecting polygon. This is an example of a systematic (and decentralizable) method for the quadcopters to identify their nearest neighbors for symmetric cyclic control. In spite of the same speed constraint of  $3m/s$ , no collisions were recorded while the formation consistently converged in terms of shape, size and geometric center.

For experimental validation, we implemented the control algorithms on a group of four quadcopters, each equipped with a PX4 Pixhawk autopilot [31]. The low level velocity tracking controller detailed in the appendix is virtually identical to the on-board Pixhawk controller. Relative position measurements between neighboring quadcopters were provided by a Vicon motion capture system. The desired inter-robot neighbor separation  $\rho$  was set to  $1.65m$ . Due to limitations in testing space, a less aggressive collision avoidance controller was implemented:

$$\mathbf{u}_{\text{coll}_i} = - \sum_{j \neq i}^n k_{\text{coll}} \tanh(\rho - d_{ij}) (\mathbf{x}_j - \mathbf{x}_i), \text{ if } d_{ij} \leq r_2,$$

where  $k_{\text{coll}}$  is a gain with value 1.2, and  $r_2 = 0.9m$ . In addition, the speed constraint was further restricted to  $0.7m/s$ , while the cyclic ( $k_1$ ) and center ( $k_c$ ) control gains were set to 0.25 and 0.15 respectively. Note that in all of the experiments where the quadcopters were initialized outside of the collision detection regions, the collision avoidance controller did not become active.

Figures 8a and 8b show the quadcopters in initial and final (tetrahedron) configurations respectively. Figure 8c shows the time evolution of the formation error  $\|\bar{\mathbf{V}}\mathbf{x}\|$  over all three phases of flight – autonomous takeoff, formation control, and landing. Figures 9a and 9b show the time evolution of the inter-quadcopter distances and the error in geometric center. The quadcopters demonstrated relatively quick convergence in terms of formation shape, size and center, thereby validating the two-level formation control methodology<sup>1</sup>.

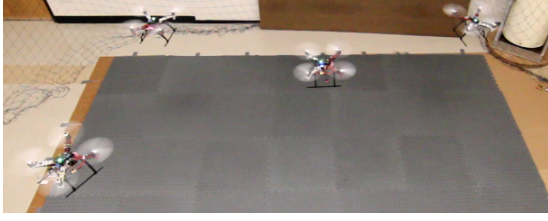
## VII. CONCLUSIONS

In this paper we studied decentralized control algorithms for 3D formations such as regular polygons, Johnson solids, and more generally, connected regular polygonal meshes. The algorithms are desirable for their simplicity in that they only require relative position measurements between neighbors and an agreement between the robots on the desired orientation in space. Our approach leveraged the mathematical properties of cyclic pursuit along with results from contraction and partial contraction theory. In addition, we provided extensions to the algorithm for controlling the formation size and preventing collisions, and quantified the robustness of the symmetric cyclic controller under bounded additive disturbances. Finally, we validated our algorithms via numerical simulations and hardware experiments on a fleet of quadcopters.

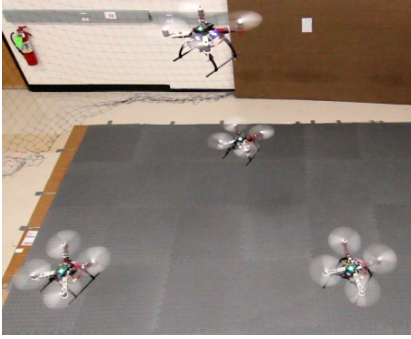
This work allows several possible avenues for extension. First, we would like to study (sub-)optimality properties of

<sup>1</sup>Videos of the experiments are available at <https://www.youtube.com/playlist?list=PL8-2mtlFIJr4fVjJeyQymbkKfhsKKYX>

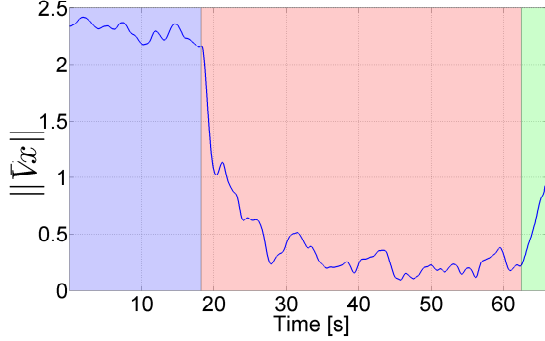




(a) Quadcopters upon initialization in hover mode.



(b) Quadcopters in final tetrahedron configuration.



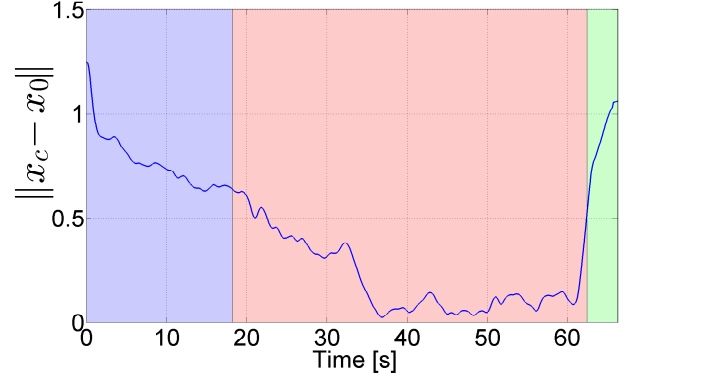
(c) Tetrahedron formation error. The three shaded regions indicate the flight modes (from left to right: takeoff, formation control, landing).

Fig. 8: Quadcopter tetrahedron formation.

the symmetric cyclic controller by introducing an objective cost function to be minimized (e.g., time or control effort), and using it to guide gain selection and/or look-ahead horizon. Second, we plan to investigate various examples for internal dynamics  $g(x)$  and their interplay with the symmetric cyclic controller when analyzing convergence. Third, the modification of the symmetric cyclic controller for formation size is a little cumbersome in that it requires all robots to compute a shared quantity. Consequently, we would like to investigate size controllers that do not rely upon consensus techniques.

## REFERENCES

- [1] W. Sheng, Q. Yang, J. Tan, and N. Xi, "Distributed Multi-Robot Coordination in Area Exploration," *Robotics and Autonomous Systems*, vol. 54, no. 12, pp. 945 – 955, 2006.
- [2] R. Smith and F. Hadaegh, "Control topologies for deep space formation flying spacecraft," in *American Control Conference*, vol. 4, 2002, pp. 2836–2841.
- [3] F. Arrichiello, H. Heidarsson, S. Chiaverini, and G. Sukhatme, "Cooperative caging using autonomous aquatic surface vehicles," in *Proc. IEEE Conf. on Robotics and Automation*, May 2010, pp. 4763–4769.
- [4] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.



(a) Error in geometric center.

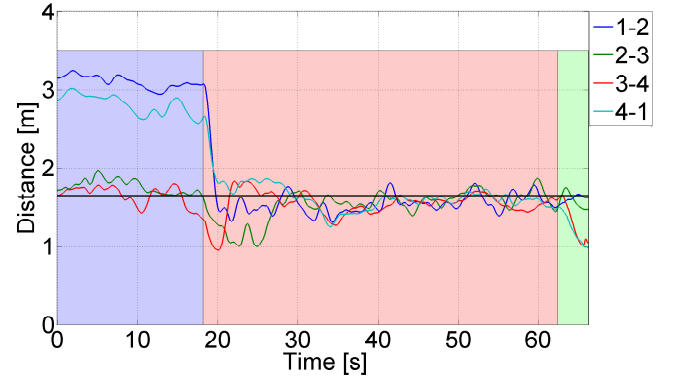
(b) Inter-quadcopter neighbor separations.  $\rho = 1.65m$ .

Fig. 9: Formation accuracy.

- [5] J. Lawton, R. Beard, and B. Young, "A decentralized approach to formation maneuvers," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 933–941, Dec 2003.
- [6] W. Ren and R. Beard, "Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 27, no. 1, pp. 73–82, Jan-Feb 2004.
- [7] J. Marshall, "Coordinated Autonomy: Pursuit Formations of Multivehicle Systems," Ph.D. dissertation, University of Toronto, 2005.
- [8] J. Chen, D. Sun, J. Yang, and H. Chen, "Leader-Follower Formation Control of Multiple Non-holonomic Mobile Robots Incorporating a Receding-horizon Scheme," *International Journal of Robotics Research*, vol. 29, no. 6, pp. 727–747, 2010.
- [9] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques, "Leader follower formation control of nonholonomic mobile robots with input constraints," *Automatica*, vol. 44, no. 5, pp. 1343 – 1349, 2008.
- [10] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, "A Survey of Spacecraft Formation Flying Guidance and Control, Part II: Control," in *American Control Conference*, vol. 4, Boston, MA, Jun. 2004, pp. 2976–2985.
- [11] Y. Cao, W. Yu, W. Ren, and G. Chen, "An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, Feb 2013.
- [12] F. Guerriero, R. Surace, V. Loscri, and E. Natalizio, "A multi-objective approach for unmanned aerial vehicle routing problem with soft time windows constraints," *Applied Mathematical Modelling*, vol. 38, no. 3, pp. 839–852, 2014.
- [13] J. Marshall, M. Broucke, and B. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Transactions on Automatic Control*, vol. 49, no. 11, pp. 1963–1974, Nov 2004.
- [14] M. Pavone and E. Frazzoli, "Decentralized Policies for Geometric Pattern Formation and Path Coverage," *ASME Journal on Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 633–643, Oct. 2007. [Online]. Available: <http://dynamicsystems.asmedigitalcollection.asme.org/article.aspx?articleid=1412612>
- [15] J. L. Ramirez, M. Pavone, E. Frazzoli, and D. W. Miller, "Distributed Control of Spacecraft Formations via Cyclic Pursuit: Theory and Experiments," *AIAA Journal of Guidance, Control, and Dynamics*,

vol. 33, no. 5, pp. 1655–1669, Sep. 2010. [Online]. Available: <http://arc.aiaa.org/doi/abs/10.2514/1.46511>

- [16] J. Ramirez-Riberos and J. Slotine, "Contraction theory approach to generalized decentralized cyclic algorithms for global formation acquisition and control," in *Proc. IEEE Conf. on Decision and Control*, Dec 2012.
- [17] W. Lohmiller and J. Slotine, "On Contraction Analysis for Non-linear Systems," *Automatica*, vol. 34, no. 6, pp. 683 – 696, 1998.
- [18] Q. Pham, N. Tabareau, and J. Slotine, "A Contraction Theory Approach to Stochastic Incremental Stability," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 816–820, April 2009.
- [19] W. Wang and J. Slotine, "On partial contraction analysis for coupled nonlinear oscillators," *Biological Cybernetics*, vol. 92, no. 1, pp. 38–53, 2005.
- [20] Q. Pham and J. Slotine, "Stable concurrent synchronization in dynamic system networks," *Neural Networks*, pp. 62–77, 2007.
- [21] O. Wallner, K. Ergenzinger, and R. Flatscher, "X-array aperture configuration in planar or non-planar spacecraft formation for Darwin/TPF-I candidate architectures," in *Proceedings of SPIE 6693, Techniques and Instrumentation for Detection of Exoplanets III*, Sep 2007.
- [22] S. Singh, E. Schmerling, and M. Pavone, "Decentralized Algorithms for 3D Symmetric Formations in Robotic Networks - A Contraction Theory Approach," in *Proc. IEEE Conf. on Robotics and Automation*, Seattle, WA, May 2015.
- [23] J. Jouffroy and J. Slotine, "Methodological remarks on contraction theory," in *Proc. IEEE Conf. on Decision and Control*, vol. 3, Dec 2004, pp. 2537–2543 Vol.3.
- [24] A. Alexandrov, *Convex Polyhedra*, 1st ed. Springer-Verlag Berlin Heidelberg, 2005, english translation with additional material of the 1950 Russian monograph.
- [25] M. J. Wenninger, *Polyhedron models*. Cambridge University Press, 1974.
- [26] N. A. Lynch, *Distributed Algorithms*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1996.
- [27] D. H. Kim, H. Wang, and S. Shin, "Decentralized control of autonomous swarm systems using artificial potential functions: Analytical design guidelines," *Journal of Intelligent and Robotic Systems*, vol. 45, no. 4, pp. 369–394, 2006.
- [28] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.
- [29] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics and Automation Magazine*, no. 19, pp. 20–32, 2012.
- [30] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.
- [31] Pixhawk. Px4 autopilot project. [Online]. Available: <http://pixhawk.org>
- [32] R. Gray, "Toeplitz and Circulant Matrices: A review," Tech. Rep., 2001.



**Dr. Marco Pavone** is an Assistant Professor of Aeronautics and Astronautics at Stanford University, where he is the Director of the Autonomous Systems Laboratory. Before joining Stanford, he was a Research Technologist within the Robotics Section at the NASA Jet Propulsion Laboratory. He received a Ph.D. degree in Aeronautics and Astronautics from the Massachusetts Institute of Technology in 2010. Dr. Pavone's areas of expertise lie in the fields of controls and robotics. His main research interests are in the development of methodologies for the analysis, design, and control of autonomous systems, with an emphasis on autonomous aerospace vehicles and large-scale robotic networks. He is a recipient of an NSF CAREER Award, a NASA Early Career Faculty Award, a Hellman Faculty Scholar Award, and was named NASA NIAC Fellow in 2011. He is currently serving as an Associate Editor for the IEEE Control Systems Magazine.



**Sumeet Singh** is a Ph.D. candidate in the Autonomous Systems Laboratory at Stanford University. He received a B.Eng. in Mechanical Engineering and a Diploma of Music (Performance) from the University of Melbourne in 2012, and a M.Sc. in Aeronautics and Astronautics from Stanford University in 2015. Prior to joining Stanford, Sumeet worked in the Berkeley Micromechanical Analysis and Design lab at University of California Berkeley in 2011, and the Aeromechanics Branch at NASA Ames in 2013. Sumeet's current research

interests include decentralized algorithms for formation control, and stochastic and robust nonlinear model predictive control, with specific applications to spacecraft control.



**Edward Schmerling** received the B.Sc. degree in mathematics and physics from Stanford University, Stanford, CA, USA, in 2010. He is currently working towards the Ph.D. degree at the Institute for Computational and Mathematical Engineering, Stanford University. His research interests include developing algorithms for robotic motion planning subject to general dynamic and cost constraints.



## APPENDIX

## VIII. MATHEMATIC PRELIMINARIES

In this section we present the eigendecomposition of circulant matrices. The following theorem will be instrumental to the proof for Theorem III.5.

**Theorem VIII.1** (Adapted from Theorem 3.1 in [32]). *The eigenvalues of a circulant matrix are the Discrete Fourier Transform of the first row of the matrix. Specifically, every circulant matrix  $C = \text{circ}[c_1 \ c_2 \ \dots \ c_n]$  has eigenvectors:*

$$\mathbf{v}_k = \frac{1}{\sqrt{n}}(1, e^{-2\pi j(k-1)/n}, \dots, e^{-2\pi j(k-1)(n-1)/n})^T, \quad (41)$$

$$k = 1, 2, \dots, n,$$

and corresponding eigenvalues:

$$\lambda_k = \sum_{p=1}^n c_p e^{2\pi j(k-1)(p-1)/n}, \quad (42)$$

and can be expressed in the form  $C = U\Lambda U^*$  where  $U$  is a unitary matrix with the  $k^{\text{th}}$  column equal to the  $k^{\text{th}}$  eigenvector and  $\Lambda$  is a diagonal matrix of corresponding eigenvalues. Note that the same matrix  $U$  diagonalizes all circulant matrices. Thus if  $C$  and  $B$  are  $n \times n$  circulant matrices, with eigenvalues  $\{\lambda_{B,k}\}_{k=1}^n$  and  $\{\lambda_{C,k}\}_{k=1}^n$  respectively, then:

- 1)  $C$  and  $B$  commute ( $CB = BC$ ) and  $CB$  is also a circulant matrix with eigenvalues equal to  $\{\lambda_{B,k}\lambda_{C,k}\}_{k=1}^n$
- 2)  $C+B$  is also circulant with eigenvalues  $\{\lambda_{B,k} + \lambda_{C,k}\}_{k=1}^n$

## IX. PROOFS

## A. Proofs for Section III

*Proof of Lemma III.1.* Necessity is trivial. We then consider sufficiency. First, we prove that constraints (4a) and (4b) ensure that all robots lie in a common plane, i.e.,

$$\mathbf{e}_z^T \mathbf{x}_i = a, \quad (43)$$

for all  $i \in \{1, \dots, n\}$  and some  $a \in \mathbb{R}$ . Since  $\mathbf{e}_z$  is a left eigenvector of  $R_{2\pi/n}$ , with eigenvalue equal to 1, the rotational constraints imply  $\mathbf{e}_z^T(\mathbf{x}_{i+1} - \mathbf{x}_i) = \mathbf{e}_z^T(\mathbf{x}_{i+2} - \mathbf{x}_{i+1})$ , for  $i = 1, \dots, n-2$ . Combining this set of equations with the in-plane constraint, one can write

$$\mathbf{e}_z^T(\mathbf{x}_{i+1} - \mathbf{x}_i) = \mathbf{e}_z^T(\mathbf{x}_{i+2} - \mathbf{x}_{i+1}),$$

for  $i = 1, \dots, n-1$  and where the indices are modulo  $n$ . One can then readily show that

$$\mathbf{e}_z^T \mathbf{x}_{i+2} = \mathbf{e}_z^T \mathbf{x}_2 + i \mathbf{e}_z^T(\mathbf{x}_2 - \mathbf{x}_1), \quad (44)$$

for  $i = 1, \dots, n-1$  and where the indices are modulo  $n$ . Hence, for  $i = n-1$ , one obtains

$$\mathbf{e}_z^T \mathbf{x}_1 = \mathbf{e}_z^T \mathbf{x}_2 + (n-1) \mathbf{e}_z^T(\mathbf{x}_2 - \mathbf{x}_1),$$

which implies that  $\mathbf{e}_z^T(\mathbf{x}_2 - \mathbf{x}_1) = \mathbf{0}$ . Hence, by setting  $\mathbf{e}_z^T \mathbf{x}_2 := a$ , one immediately obtains equation (43). We have then proven that all robots lie in a common plane. Now, for  $n$  points lying in a common plane, the rotational constraints represent the definition of a regular polygon.

For independence, we note that each successive rotational constraint involves at least one robot that does not belong to any of the previous constraints, proving that the rotational constraints are independent. The in-plane constraint is necessary

to ensure that the robots do not form a spiral formation out of the desired plane, thereby reducing the dimensionality of the solution subspace. Thus, the in-plane constraint must be independent to the rotational constraints.  $\square$

*Proof of Lemma III.3.* Equations (4a) and (4b) can be written in matrix form as  $V \mathbf{x} = \mathbf{0}$ , where

$$V = \begin{bmatrix} I_3 & -(I_3 + R_{2\pi/n}) & R_{2\pi/n} & & \\ 0_{3 \times 3} & I_3 & -(I_3 + R_{2\pi/n}) & & \\ \vdots & \vdots & \vdots & \ddots & \\ \mathbf{e}_z^T R_{2\pi/n} & 0_{1 \times 3} & 0_{1 \times 3} & & \\ 0_{3 \times 3} & \cdots & \cdots & 0_{3 \times 3} & \\ R_{2\pi/n} & 0_{3 \times 3} & \cdots & \vdots & \\ \vdots & \vdots & \vdots & \vdots & \\ \cdots & \cdots & \mathbf{e}_z^T & -\mathbf{e}_z^T(I_3 + R_{2\pi/n}) & \end{bmatrix}. \quad (45)$$

Simplifying the above expression, one obtains

$$\begin{aligned} V &= \mathcal{W}_n \text{circ}[I_3, -I_3, 0_{3 \times 3}, \dots, 0_{3 \times 3}] + \\ &\quad \mathcal{W}_n \text{circ}[0_{3 \times 3}, -R_{2\pi/n}, R_{2\pi/n}, 0_{3 \times 3}, \dots, 0_{3 \times 3}] \\ &= \mathcal{W}_n(L_1 \otimes I_3) + \mathcal{W}_n((L_1 - L_2) \otimes R_{2\pi/n} I_3) \\ &= \mathcal{W}_n(L_1 I_n \otimes I_3 I_3 + (L_1 - L_2) I_n \otimes R_{2\pi/n} I_3) \\ &= \mathcal{W}_n(L_1 \otimes I_3 + (L_1 - L_2) \otimes R_{2\pi/n})(I_n \otimes I_3), \end{aligned} \quad (46)$$

where the last equality follows from the properties of the Kronecker product. The claim then follows.  $\square$

*Proof of Lemma III.4.* Note that the rotational constraints in (4a) and the in-plane constraint (4b) define the subspace  $\mathcal{M}_n$  to be a regular polygon with normal  $\mathbf{e}_z$ . The following analysis therefore assumes that all robots lie in a single plane and satisfy the rotational constraints. Then,

$$\begin{aligned} \dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i &= \sum_{m=1}^N k_m(\mathbf{x}, t) [R_m(\mathbf{x}, t)(\mathbf{x}_{i+1+m} - \mathbf{x}_{i+1}) \\ &\quad + R_m^T(\mathbf{x}, t)(\mathbf{x}_{i+1-m} - \mathbf{x}_{i+1})] \\ &\quad - \sum_{m=1}^N k_m(\mathbf{x}, t) [R_m(\mathbf{x}, t)(\mathbf{x}_{i+m} - \mathbf{x}_i) \\ &\quad + R_m^T(\mathbf{x}, t)(\mathbf{x}_{i-m} - \mathbf{x}_i)]. \end{aligned} \quad (47)$$

Now by the constraints described by (4a), we can state:

$$\begin{aligned} \mathbf{x}_{i+1+m} - \mathbf{x}_{i+1} &= R_{2\pi/n}(\mathbf{x}_{i+2+m} - \mathbf{x}_{i+2}), \text{ and} \\ \mathbf{x}_{i+1-m} - \mathbf{x}_{i+1} &= R_{2\pi/n}(\mathbf{x}_{i+2-m} - \mathbf{x}_{i+2}). \end{aligned}$$

Thus, the first term in (47) can be written as:

$$\begin{aligned} \dot{\mathbf{x}}_{i+1} &= \sum_{m=1}^N k_m(\mathbf{x}, t) [R_m R_{2\pi/n}(\mathbf{x}, t)(\mathbf{x}_{i+2+m} - \mathbf{x}_{i+2}) \\ &\quad + R_m^T R_{2\pi/n}(\mathbf{x}, t)(\mathbf{x}_{i+2-m} - \mathbf{x}_{i+2})] \\ &= R_{2\pi/n} \sum_{m=1}^N k_m(\mathbf{x}, t) [R_m(\mathbf{x}, t)(\mathbf{x}_{i+2+m} - \mathbf{x}_{i+2}) \\ &\quad + R_m^T(\mathbf{x}, t)(\mathbf{x}_{i+2-m} - \mathbf{x}_{i+2})] \\ &= R_{2\pi/n} \dot{\mathbf{x}}_{i+2}, \end{aligned} \quad (48)$$

where the order of  $R_m$  and  $R_{2\pi/n}$  can be swapped since they are rotation matrices about the same axis. Similarly, it can be shown that  $\dot{\mathbf{x}}_i = R_{2\pi/n}\dot{\mathbf{x}}_{i+1}$ . Thus, rewriting (47):

$$\dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i = R_{2\pi/n}(\dot{\mathbf{x}}_{i+2} - \dot{\mathbf{x}}_{i+1}).$$

To prove the arbitrary normal case, one need only replace  $R_m$  and  $R_{2\pi/n}$  with their similarity transformed versions with respect to  $R_\eta \neq I_3$ .  $\square$

*Proof of Theorem III.5.* We provide the proof for the general case where the desired formation plane is not necessarily the horizontal plane. The proof leverages the following Lemma.

**Lemma IX.1.** *Given a symmetric matrix  $\mathcal{X}$  such that  $\mathcal{X} \prec 0$ , then  $\mathcal{W}_n \mathcal{X} \mathcal{W}_n^T \prec 0$ .*

*Proof.* If  $\mathcal{X} \prec 0$  then  $\lambda_{\max}(\mathcal{X}) < 0$ . Noting that  $\mathcal{W}_n$  is sub-unitary, by the Cauchy Interlacing Theorem,  $\lambda_{\max}(\mathcal{W}_n \mathcal{X} \mathcal{W}_n^T) \leq \lambda_{\max}(\mathcal{X}) < 0$ .  $\square$

Having proved Lemma IX.1, we are now in a position to prove Theorem III.5. Specifically, the strategy is to only consider the uniform negative-definiteness of the inner term (not involving  $\mathcal{W}_n$ ) in (8).

For the scenario where the desired formation plane is not the horizontal plane, by the discussion in Remark III.7, we must use the similarity transformed versions of the projection matrix  $V$  and the symmetric cyclic controller. From (10) and (11), note that  $\mathcal{P}_n \mathcal{R}_\eta \mathcal{L}_{m\eta} \mathcal{R}_\eta^T \mathcal{P}_n^T = \mathcal{P}_n \mathcal{L}_m \mathcal{P}_n^T$  since  $\mathcal{L}_{m\eta} = \mathcal{R}_\eta^T \mathcal{L}_m \mathcal{R}_\eta$ . Thus, the analysis does not depend on the arbitrary choice of the orientation of the desired plane normal in the global coordinate system.

Now, given the matrices  $\mathcal{P}_n$  and  $\mathcal{L}_m \in \mathcal{C}_{L \otimes R}$ , they have the same set of eigenvectors. Then, for any eigenvector  $\mathbf{v}_i$  in this set, we have the following relation for the corresponding eigenvalues:

$$\lambda_i(\mathcal{P}_n \mathcal{L}_m \mathcal{P}_n^T) = \lambda_i(\mathcal{P}_n) \lambda_i(\mathcal{P}_n^T) \lambda_i(\mathcal{L}_m).$$

To derive the eigenvalues of  $\mathcal{P}_n$ , we note that  $\mathcal{P}_n = L_1 \otimes I_3 + (L_1 - L_2) \otimes R_{2\pi/n}$  which is the sum of two  $\mathcal{C}_{L \otimes R}$  matrices and thus the eigenvalues of  $\mathcal{P}_n$  must be the sum of the eigenvalues of  $L_1 \otimes I_3$  and  $(L_1 - L_2) \otimes R_{2\pi/n}$ . From Theorem VIII.1, we have the general result:

$$\lambda_i(L_m) = 1 - e^{\frac{2m\pi}{n}(i-1)j}, \quad i = 1, \dots, n. \quad (49)$$

Additionally, from Theorem VIII.1, the eigenvalues of  $L_1 - L_2$  are the sum of eigenvalues of  $L_1$  and  $-L_2$ . The eigenvalues of  $R_{2\pi/n}$  are  $\{1, e^{\pm \frac{2\pi}{n}j}\} = e^{\pm \frac{2\pi}{n}j}$  where  $k \in \{-1, 0, 1\}$ . Thus, the eigenvalues of  $\mathcal{P}_n$  and  $\mathcal{P}_n^T$  are the set  $i \in \{1, \dots, n\}, k \in \{-1, 0, 1\}$ :

$$\lambda(\mathcal{P}_n, \mathcal{P}_n^T) = \left(1 - e^{\pm \frac{2\pi}{n}(i-1)j}\right) + \left(e^{\pm \frac{4\pi}{n}(i-1)j} - e^{\pm \frac{2\pi}{n}(i-1)j}\right) e^{\pm \frac{2\pi}{n}j}, \quad (50)$$

where the positive and negative signs are for  $\mathcal{P}_n$  and  $\mathcal{P}_n^T$  respectively. In a similar fashion, the eigenvalues for  $\mathcal{L}_m$  are:

$$\begin{aligned} \lambda(\mathcal{L}_m) &= \left(1 - e^{\frac{2m\pi}{n}(i-1)j}\right) e^{k\alpha_m j} \\ &\quad + \left(1 - e^{\frac{-2m\pi}{n}(i-1)j}\right) e^{-k\alpha_m j} \\ &= 2 \left( \cos(k\alpha_m) - \cos\left(k\alpha_m + \frac{2\pi m(i-1)}{n}\right) \right). \end{aligned} \quad (51)$$

Multiplying (50) and (51) gives the expression for  $\lambda(\mathcal{P}_n \mathcal{L}_m \mathcal{P}_n^T) = \lambda_{ik}^{(m)}(\mathbf{x}, t)$ . To obtain the summation form in (9), note that  $\mathcal{P}_n \mathcal{L}_m \mathcal{P}_n^T \in \mathcal{C}_{L \otimes R}$  for all  $m$ . Thus  $\lambda\left(\mathcal{P}_n \sum_{m=1}^N k_m \mathcal{L}_m \mathcal{P}_n^T\right) = \sum_{m=1}^N k_m [\lambda(\mathcal{P}_n \mathcal{L}_m \mathcal{P}_n^T)]$ . Having obtained the summation form, (9) follows directly from trying to show (8) given Lemma IX.1.  $\square$

## B. Proofs for Section IV

*Proof of Corollary IV.5.* Necessity is straightforward. We then consider sufficiency. Let us assume without loss of generality that  $\mathbf{n}_k = \mathbf{e}_z$ ; the general case follows through a rotation under  $R_{\eta_k}$ . To prove the claim, we need to show that the rotational constraints are sufficient to ensure that all robots lie in the desired plane for  $j = 1, \dots, |\mathcal{V}_k|$ . The claim then follows directly. Now, the rotational constraints imply

$$\mathbf{e}_z^T(\mathbf{x}_{i+1} - \mathbf{x}_i) = \mathbf{e}_z^T(\mathbf{x}_{i+2} - \mathbf{x}_{i+1}), \quad (52)$$

for  $i = 1, \dots, |\mathcal{V}_k| - 2$ . For two robots with indices  $j$  and  $j+1$  that lie in the desired plane, we know that  $\mathbf{e}_z^T \mathbf{x}_j = \mathbf{e}_z^T \mathbf{x}_{j+1} = a \in \mathbb{R}$ . If  $j \in \{1, \dots, |\mathcal{V}_k| - 1\}$ , we can recursively use (52) to show

$$\mathbf{e}_z^T \mathbf{x}_i = \mathbf{e}_z^T \mathbf{x}_j, \quad (53)$$

for  $i = 1, \dots, |\mathcal{V}_k|$ . For the case where  $j = |\mathcal{V}_k|$ , we have:

$$\mathbf{e}_z^T \mathbf{x}_n = \mathbf{e}_z^T \mathbf{x}_1.$$

However, we note that (44) still holds for  $i = 1, \dots, |\mathcal{V}_k| - 2$  using the rotational constraints alone. Then, setting  $i = |\mathcal{V}_k| - 2$ , we obtain

$$(\mathbf{e}_z^T \mathbf{x}_n - \mathbf{e}_z^T \mathbf{x}_1) + (|\mathcal{V}_k| - 1) \mathbf{e}_z^T (\mathbf{x}_2 - \mathbf{x}_1) = (|\mathcal{V}_k| - 1) \mathbf{e}_z^T (\mathbf{x}_2 - \mathbf{x}_1),$$

which gives  $\mathbf{e}_z^T \mathbf{x}_2 = \mathbf{e}_z^T \mathbf{x}_1$ . We can now again use (52) recursively to show (53), completing the proof.  $\square$

## C. Proofs for Section V

*1) Control Over Formation Size:* In order to prove Theorem V.2, we first require several intermediate results detailed in the following lemmas. We first prove that  $\mathbf{x}$  converges to the subspace  $\mathcal{M}_n$ .

**Lemma IX.2** (Convergence to  $\mathcal{M}_n$  with Size Control). *Assume*

$$\inf_{\alpha_m \in [m\pi/n - \alpha_{s_0}, m\pi/n + \alpha_{s_0}]} \left( \min_{\substack{1 \leq i \leq n \\ k \in \{-1, 0, 1\}}} \sum_{m=1}^N k_m \lambda_{ik}^{(m)} \right) > 0, \quad (54)$$

where  $\lambda_{ik}^{(m)}$  has the form given in Theorem III.5. Then the system  $\dot{\mathbf{x}} = \mathbf{u}$  under the controller (26) globally converges to  $\mathcal{M}_n$ .

*Proof.* To show convergence to  $\mathcal{M}_n$ , consider the following auxiliary system for the transformed closed-loop dynamics  $\dot{\mathbf{z}}_p = \bar{V} \dot{\mathbf{x}}_p = \bar{V} \mathbf{u}$  where  $\mathbf{u}$  is as given in (26):

$$\dot{\mathbf{y}} = -\bar{V} \mathcal{L}(\mathbf{x}_p, \tau) (\bar{V}^T \mathbf{y} + \bar{U}^T \bar{U} \mathbf{x}_p),$$

Note that  $\mathbf{y} = \mathbf{0}$  is a solution of this auxiliary system since  $\bar{V}\mathcal{L}(\mathbf{x}_p, \tau)(\bar{U}^T \bar{U} \mathbf{x}_p) = \mathbf{0}$ ,  $\forall \mathbf{x}_p$ . Then to show that the auxiliary system is contracting in  $\mathbf{y}$ , we require:

$$\bar{V}(\mathcal{L}(\mathbf{x}_p, \tau))\bar{V}^T \succ 0, \quad \text{uniformly, } \forall \mathbf{x}_p, \tau.$$

By Remark II.4, the above stability requirement can be reformulated in terms of  $V$  directly, i.e.,

$$\mathcal{W}_n \mathcal{P}_n(\mathcal{L}(\mathbf{x}_p, \tau))\mathcal{P}_n^T \mathcal{W}_n^T \succ 0, \quad \text{uniformly, } \forall \mathbf{x}_p, \tau. \quad (55)$$

The above condition is equivalent to (8) with  $\mathbf{g} = \mathbf{0}$ ,  $\mathcal{L} = \mathcal{L}(\mathbf{x}_p, \tau)$  and the additional stipulation that (55) holds for all possible  $\mathbf{x}_p$  and time lags  $\tau$ . Temporarily neglecting this additional condition and applying Theorem III.5 with  $\mathbf{g} = \mathbf{0}$  gives the expression within the infimum in (54). Then to account for all possible  $\mathbf{x}_p$  and time lags  $\tau$ , we note that the dependance of  $\mathcal{L}$  on  $\mathbf{x}_p$  and  $\tau$  stems from the rotation angle  $\alpha_m = m\pi/n + \alpha_s(\mathbf{x}_p, \tau)$ , which changes every  $\tau$  seconds. As  $|f_s(\bar{p})| \leq 1$ , it follows that  $\alpha_m \in [m\pi/n - \alpha_{s0}, m\pi/n + \alpha_{s0}]$ , for all time (despite the discrete switches every  $\tau$  seconds). Choosing the worst case value for  $\alpha_m$  within this interval then completes the expression given in (54).  $\square$

**Remark IX.3.** Note that the continuous/discrete hybrid dynamics induced by the time lag parameter  $\tau$  do not affect the convergence analysis for  $\mathcal{M}_n$ . By taking the infimum in (54), we have that for any modification of the standard rotation angle  $m\pi/n$  by the angle  $\alpha_s$ , condition (55) is still satisfied.

We now study the behavior of the system within the invariant subspace  $\mathcal{M}_n$ , with respect to the desired subset  $\mathcal{M}_{n\rho}$ . Here, the time lag  $\tau$  will play a crucial role in governing the convergence of the inter-robot separation to the desired distance.

**Lemma IX.4** (Formation Size Discrete Dynamics). Assume  $\mathbf{x}$  lies in the subspace  $\mathcal{M}_n$ . Let  $\beta = \sqrt{2(1 - \cos(2\pi/n))}$ ,  $\Gamma = \sum_{m=1}^N k_m \gamma_m$  where  $\gamma_m$  are fixed constants derivable from geometry, specifically,

$$\gamma_1 = 1, \text{ and } \gamma_m \in \begin{cases} \left(1, \frac{1}{\sin(\frac{\pi}{n})}\right] & \text{if } n \text{ is even,} \\ \left(1, \frac{1}{2\sin(\frac{\pi}{2n})}\right] & \text{if } n \text{ is odd,} \end{cases}$$

for  $m = 2, \dots, N$ , and let  $C = 2\beta\Gamma$ . Then, under the action of the modified cyclic controller given in (26), each inter-robot distance error  $p_i$  admits the following recursive solution:

$$p_{i,k+1} = (p_{i,k} - 1)e^{C \sin(\alpha_s(p_{i,k-1}))\tau} + 1, \quad (56)$$

where  $p_{i,k} = p_i(k\tau)$ .

*Proof.* First note that by Lemma V.1,  $\mathcal{M}_n$  is flow-invariant with respect to the control law given in (26). Thus, we will exploit the symmetry properties attributed to regular polygons. In particular, recall the definition  $p_i := 1 - \|\mathbf{x}_{i+1} - \mathbf{x}_i\|/\rho$ . We note that for  $\mathbf{x} \in \mathcal{M}_n$ ,  $p_1 = p_2 = \dots = p_n = \bar{p}$ . Consider the time-derivative of  $p_i$  within the time-range  $[k\tau, (k+1)\tau)$ :

$$\dot{p}_i = -\frac{(\mathbf{x}_{i+1} - \mathbf{x}_i)^T(\dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i)}{\rho\|\mathbf{x}_{i+1} - \mathbf{x}_i\|}.$$

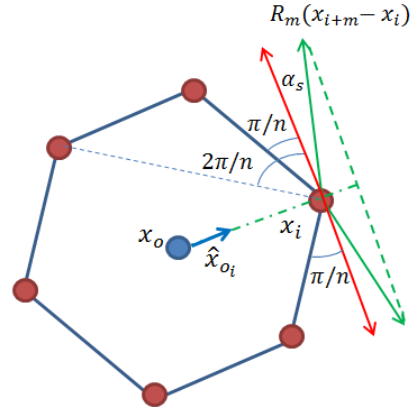


Fig. 10: Cyclic control with  $\alpha_s$

To evaluate  $(\dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i)$ , consider Figure 10.

From Remark III.6, we see the geometric realization of the forward and rear components of the cyclic controller cancelling for  $\mathbf{x} \in \mathcal{M}_n$ , when  $\alpha_m = m\pi/n$ . We can also see that the vector  $R_m(\mathbf{x}_{i+m} - \mathbf{x}_i)$  where  $R_m$  is a rotation matrix with angle  $m\pi/n + \alpha_s$ , is simply a rotation of  $R_{m\pi/n}(\mathbf{x}_{i+m} - \mathbf{x}_i)$  about the plane normal by the angle  $\alpha_s$ . Thus, we can write:

$$\begin{aligned} \dot{\mathbf{x}}_i &= 2 \sin(\alpha_s) \left( \sum_{m=1}^N k_m \|\mathbf{x}_{i+m} - \mathbf{x}_i\| \right) \hat{\mathbf{x}}_{o_i} \\ \dot{\mathbf{x}}_{i+1} &= 2 \sin(\alpha_s) \left( \sum_{m=1}^N k_m \|\mathbf{x}_{i+1+m} - \mathbf{x}_{i+1}\| \right) \hat{\mathbf{x}}_{o_{i+1}} \\ &= 2 \sin(\alpha_s) \left( \sum_{m=1}^N k_m \|\mathbf{x}_{i+m} - \mathbf{x}_i\| \right) \hat{\mathbf{x}}_{o_{i+1}}, \end{aligned}$$

where  $\hat{\mathbf{x}}_{o_i}$  is the unit vector pointing outward from the instantaneous geometric center  $\mathbf{x}_0$  towards robot  $i$ . Exploiting symmetry, we can re-write  $\|\mathbf{x}_{i+m} - \mathbf{x}_i\|$  as  $\gamma_m \|\mathbf{x}_{i+1} - \mathbf{x}_i\|$ , where  $\gamma_1 = 1$  and  $\gamma_m$  for  $m = 2, \dots, N$  may be derived from geometry. Using these simplifications, we obtain

$$\begin{aligned} (\dot{\mathbf{x}}_{i+1} - \dot{\mathbf{x}}_i) &= 2 \sin(\alpha_s) \left( \sum_{m=1}^N k_m \gamma_m \right) \|\mathbf{x}_{i+1} - \mathbf{x}_i\| \\ &\quad \times \underbrace{\sqrt{2(1 - \cos(2\pi/n))}}_{:=\beta>0} \frac{(\mathbf{x}_{i+1} - \mathbf{x}_i)}{\|\mathbf{x}_{i+1} - \mathbf{x}_i\|}. \end{aligned}$$

We now recall that  $\alpha_s = \alpha_s(\bar{p}_{k-1})$ , and is constant over the time range  $[k\tau, (k+1)\tau)$ . Thus, we can write  $\dot{p}_i$  as:

$$\begin{aligned} \dot{p}_i &= -2\beta\Gamma \sin(\alpha_s(\bar{p}_{k-1})) \frac{\|\mathbf{x}_{i+1} - \mathbf{x}_i\|}{\rho} \\ &= C \sin(\alpha_s(p_{i,k-1}))(p_i - 1). \end{aligned} \quad (57)$$

Solving the above equation exactly gives the recursive formula given in (56).  $\square$

We are now ready to derive sufficient conditions for the time lag parameter  $\tau$  to ensure that the recursive solution given in (56) converges to 0.

**Lemma IX.5** (Convergence to  $\mathcal{M}_{n\rho}$  within  $\mathcal{M}_n$ ). Assume  $\mathbf{x}$  lies in the subspace  $\mathcal{M}_n$ . Let  $T$  denote a positive constant such that  $(1/2)T|p_i| \leq |\sin(\alpha_{s_0} f_s(p_i))| \leq T|p_i|$  for  $|p_i| < 1$  (if  $f_s$  is the saturation function,  $T = \alpha_{s_0}$ ). Suppose the time lag parameter  $\tau$  satisfies the following bound:

$$\tau < \min \left\{ \frac{1}{C}, \frac{1}{8CT} \right\}. \quad (58)$$

Then the neighbor inter-robot distance governed by the discrete dynamics given in (56) converges to the desired distance  $\rho$ . That is,  $\mathbf{x}$  converges to the subspace  $\mathcal{M}_{n\rho}$ .

*Proof.* The layout of the proof is as follows: we first show that there exists some time index  $k'$  such that the inter-robot distance error  $p_{i_k}$  has the same sign in two consecutive time steps. Next, we prove that if  $p_{i_k}$  has the same sign for all  $k > k'$ , then  $p_{i_k}$  converges to zero. Finally, we provide sufficient conditions to ensure that either (a)  $p_{i_k}$  possesses the same sign for all  $k > k'$ , or (b) if there exists a time index  $k''$  where  $p_{i_k}$  changes sign, then it never crosses 0 again. We begin with the following claim:

**Claim IX.6.** Given the inter-robot distance error discrete dynamics given in (56), there exists a time index  $k'$  such that  $p_{i_{k'-1}}p_{i_{k'}} > 0$ . That is, the distance error has the same sign in two consecutive time steps.

*Proof.* Assume by way of contradiction that  $p_{i_{k-1}}p_{i_k} < 0$ ,  $p_{i_{k-1}} < 0$  and  $1 > p_{i_k} > 0$ . Then,  $\sin(\alpha_s(p_{i_{k-1}})) < 0$ . From (56), we have  $1 > p_{i_{k+1}} > p_{i_k} > 0$ . Alternatively, assume that  $p_{i_{k-1}}p_{i_k} < 0$ ,  $p_{i_{k-1}} > 0$  and  $p_{i_k} < 0$ . Then,  $\sin(\alpha_s(p_{i_{k-1}})) > 0$ . Using (56), we have  $p_{i_{k+1}} < p_{i_k} < 0$ . Setting  $k' = k + 1$  gives the desired contradiction.  $\square$

Given there exists a time index  $k'$ , such that  $p_{i_{k'-1}}p_{i_{k'}} > 0$ , consider the following result:

**Claim IX.7.** Let  $k'$  be a time index such that  $p_{i_{k'-1}}p_{i_{k'}} > 0$ . If

- 1)  $p_{i_{k'}} > 0$  and  $p_{i_k} > 0$  for all  $k > k'$ , or
- 2)  $p_{i_{k'}} < 0$  and  $p_{i_k} < 0$  for all  $k > k'$ ,

then  $p_{i_k}$  converges to 0.

*Proof.* Consider the first case. Without loss of generality, assume  $p_{i_{k'-1}} > p_{i_{k'}}$ . If this is not true, then take  $k''$  to be  $k' + 1$ . Then given (56), we have  $p_{i_{k''}} = p_{i_{k'+1}} < p_{i_{k'}} = p_{i_{k''-1}}$ , and we proceed with our analysis for  $k > k''$ .

Now if  $p_{i_k} > 0$  for all  $k > k'$ , and  $p_{i_k}p_{i_{k-1}} > 0$  for all  $k > k'$ , then given (56), we deduce that  $p_{i_k}$  is a strictly decreasing monotonic sequence lower-bounded by 0, and therefore has a limit  $l \geq 0$ . We will show that  $l = 0$ .

Consider the continuous time differential equation in (57). Given the recursive solution in (56), we know that  $p_i(t) < p_{i_{k'-1}}$  (and consequently  $\alpha_s(p_i(t)) < \alpha_s(p_{i_{k'-1}})$ ) for all  $t > (k' - 1)\tau$ . Furthermore, since  $p_i(t) - 1 < 0$ , we obtain the following differential inequality for  $t \in [k'\tau, (k' + 1)\tau]$ :

$$\begin{aligned} \dot{p}_i(t) &= C \sin(\alpha_s(p_{i_{k'-1}})) (p_i(t) - 1) \\ &< C \sin(\alpha_s(p_i(t))) (p_i(t) - 1). \end{aligned} \quad (59)$$

We now use the lower bound:  $\sin(\alpha_s(p_i(t))) > (1/2)T p_i(t)$ . It follows then:

$$\dot{p}_i(t) < \frac{CT}{2} p_i(t) (p_i(t) - 1).$$

By the comparison theorem [28], we obtain for  $t \in [k'\tau, (k' + 1)\tau]$ :

$$p_i(t) < \frac{1}{1 + \left( \frac{1}{p_{i_{k'}}} - 1 \right) e^{\frac{CT}{2}(t - k'\tau)}} := \xi(t).$$

In particular, we note that

$$p_{i_{k'+1}} < \xi((k' + 1)\tau).$$

Proceeding inductively, it follows that

$$p_{i_k} < \xi(k\tau), \forall k > k'.$$

That is, the recursive solution is bounded above by  $\xi(t)$ , a Class- $\mathcal{L}$  function [28], for all  $k > k'$ . Consequently, the limit  $l$  for  $p_{i_k}$  must equal 0.

Consider now the second case. Without loss of generality, assume  $p_{i_{k'}} > p_{i_{k'-1}}$ . If this is not true, then take  $k''$  to be  $k' + 1$ . Then given (56), we have  $p_{i_{k''}} = p_{i_{k'+1}} > p_{i_{k'}} = p_{i_{k''-1}}$ , and we proceed with our analysis for  $k > k''$ .

Now if  $p_{i_k} < 0$  for all  $k > k'$  and  $p_{i_k}p_{i_{k-1}} > 0$  for all  $k > k'$ , then given (56), we deduce that  $p_{i_k}$  is a strictly increasing monotonic sequence upper-bounded by 0, and therefore has a limit  $-l \leq 0$ . We will show that  $l = 0$ .

We first prove that  $l < 1$ . That is, there exists a time index  $k^*$  such that for all  $k > k^*$ ,  $|p_{i_k}| < 1$ . By way of contradiction, assume that  $l \geq 1$  and suppose  $p_{i_k} = -l - \epsilon$  where  $\epsilon > 0$  is an infinitesimally small positive number. Given (56), we know that  $p_{i_{k-1}} < p_{i_k} < -1$ . Then since  $C\tau > 0$ , we know that  $C \sin(\alpha_s(p_{i_{k-1}}))\tau < C \sin(\alpha_{s_0} f_s(-1))\tau < 0$ , and thus  $e^{C \sin(\alpha_s(p_{i_{k-1}}))\tau} < 1$ . Since the sequence  $p_{i_k}$  is increasing monotonically, for the limit to be  $-l$ ,  $p_{i_k} < -l$  for all  $k$ . To obtain the desired contradiction, we will show that there exists a small enough  $\epsilon$  such that  $p_{i_{k+1}} > -l$ . Using (56), we seek an  $\epsilon > 0$  such that:

$$(-l - \epsilon - 1)e^{C \sin(\alpha_s(p_{i_{k-1}}))\tau} > -l - 1.$$

Re-arranging the above expression, we require:

$$e^{C \sin(\alpha_s(p_{i_{k-1}}))\tau} < \frac{l + 1}{l + 1 + \epsilon},$$

Since  $e^{C \sin(\alpha_s(p_{i_{k-1}}))\tau} < e^{C \sin(\alpha_{s_0} f_s(-1))\tau} < 1$ , there exists an  $\epsilon > 0$  small enough such that the above condition will be satisfied. We now proceed with our analysis for  $k > k^*$ . For  $t > k^*\tau$ , we have that  $|p_i(t)| < |p_{i_{k^*}}|$ . Then, using (57) and that  $|p_i(t)| < 1$ , we obtain the following differential inequality for  $t \in [(k^* + 1)\tau, (k^* + 2)\tau]$ :

$$\begin{aligned} \dot{p}_i(t) &= -C |\sin(\alpha_s(p_{i_{k^*}}))| (p_i(t) - 1) \\ &> -C |\sin(\alpha_s(p_i(t)))| (p_i(t) - 1). \end{aligned}$$

Using the lower bound:  $|\sin(\alpha_s(p_i(t)))| > (1/2)T|p_i(t)|$  for  $t > k^*\tau$ , it follows:

$$\dot{p}_i(t) > -\frac{CT}{2} |p_i(t)| (p_i(t) - 1).$$

Once again, applying the Comparison Theorem [28], we obtain for  $t \in [(k^* + 1)\tau, (k^* + 2)\tau]$ :

$$|p_i(t)| < \left| \frac{1}{1 + \left( \frac{1}{p_{i_{k^*+1}}} - 1 \right) e^{\frac{CT}{2}(t - (k^* + 1)\tau)}} \right| := \eta(t).$$

We note that:

$$|p_{i_{k^*+2}}| < \eta((k^* + 2)\tau).$$

Proceeding inductively, it follows that

$$|p_{i_k}| < \eta(k\tau), \quad \forall k > k^* + 1.$$

Thus,  $|p_{i_k}|$  is bounded above by  $\eta(t)$ , a Class- $\mathcal{L}$  function [28], for all  $k > k^* + 1$ . Thus, the limit  $-l$  for  $p_{i_k}$  must equal 0.  $\square$

Now suppose  $p_{i_{k'-1}}p_{i_{k'}} > 0$  and  $p_{i_{k'}} > 0$ . We now derive sufficient conditions to ensure that  $p_{i_k} > 0$  for all  $k > k'$ . Let  $k = k'$ . Since  $\tau < 1/C$ , and  $0 < \sin(\alpha_s(p_{i_{k-1}})) < 1$ , we have  $0 < C \sin(\alpha_s(p_{i_{k-1}}))\tau < 1$ . Using the inequality  $e^\zeta \leq 1 + 2\zeta$  for  $\zeta \in [0, 1]$ , we may deduce:

$$e^{C \sin(\alpha_s(p_{i_{k-1}}))\tau} \leq 1 + 2C \sin(\alpha_s(p_{i_{k-1}}))\tau.$$

Additionally, since  $|\sin(\alpha_{s_0} f_s(p_{i_{k-1}}))| \leq T|p_{i_{k-1}}|$ , we have:

$$1 + 2C \sin(\alpha_s(p_{i_{k-1}}))\tau \leq 1 + \underbrace{2CT\tau}_{:=A>0} p_{i_{k-1}}.$$

Since,  $p_{i_k} - 1 < 0$ , it follows

$$p_{i_{k+1}} \geq (p_{i_k} - 1)(1 + Ap_{i_{k-1}}) + 1 = p_{i_k} + Ap_{i_{k-1}}p_{i_k} - Ap_{i_{k-1}}.$$

Equivalently,

$$\frac{p_{i_{k+1}}}{p_{i_k}} \geq 1 + Ap_{i_{k-1}} - A \frac{p_{i_{k-1}}}{p_{i_k}}.$$

Note that  $p_{i_k}/p_{i_{k-1}} > 1 > 1/2$  (i.e.  $p_{i_{k-1}}/p_{i_k} < 2$ ) and  $Ap_{i_{k-1}} > 0$ . Then, if  $Ap_{i_{k-1}}/p_{i_k} < 1/2$  (i.e.  $A < 1/4$ ), then,

$$0 < (1/2)p_{i_k} < p_{i_{k+1}} < p_{i_k}.$$

By induction, we conclude that  $p_{i_k} > 0$  for all  $k > k'$ . By Claim IX.7,  $p_{i_k}$  converges to 0. The requirement  $A < 1/4$  is guaranteed by (58).

Finally, we address the scenario where  $p_{i_{k'-1}}p_{i_{k'}} > 0$  and  $p_{i_{k'}} < 0$ . There are two sub-cases:

- 1) For all  $k > k'$ ,  $p_{i_k} < 0$ . Then by Claim IX.7 above,  $p_{i_k}$  converges to 0.
- 2) There exists a time index  $k''$  such that  $p_{i_{k''-1}}p_{i_{k''}} < 0$  (i.e.  $p_{i_{k''}} > 0$ ).

Analyzing the second sub-case, using (56), we have that  $p_{i_{k''+1}} > p_{i_{k''}} > 0$ . Then given  $A < 1/4$ ,  $p_{i_k} > 0$  for all  $k > k''$ . Thus,  $p_{i_k}$  does not change sign again, completing the proof.  $\square$

We are now ready to prove Theorem V.2 which links convergence analysis both on and off the subspace  $\mathcal{M}_n$ .

*Proof of Theorem V.2.* By Lemmas V.1, IX.4 and IX.5, we know that if  $\mathbf{x}(0)$  lies in the flow-invariant subspace  $\mathcal{M}_n$ , then  $\mathbf{x}$  converges to the manifold  $\mathcal{M}_{n\rho}$ .

We now address the scenario where  $\mathbf{x}(0) \in \mathbb{R}^{3n} \setminus \mathcal{M}_n$ . The state  $\mathbf{x}$  may be decomposed into projections both on and off  $\mathcal{M}_n$  as follows:

$$\mathbf{x} = \underbrace{\bar{U}^T \bar{U} \mathbf{x}}_{:=\mathbf{x}_M \in \mathcal{M}_n} + \underbrace{\bar{V}^T \bar{V} \mathbf{x}}_{:=\mathbf{x}_{M^\perp} \in \mathcal{M}_n^\perp}.$$

By Lemma IX.2, we have that  $\|\mathbf{x}_{M^\perp}\|$  is decreasing exponentially to 0. Thus  $\mathbf{x} \rightarrow \mathbf{x}_M$ . All that remains is to show

that  $\mathbf{x}_M \rightarrow \mathcal{M}_{n\rho}$ . To prove this result, let us examine the dynamics of  $\mathbf{x}_M$ :

$$\begin{aligned} \dot{\mathbf{x}}_M &= \bar{U}^T \bar{U} \mathbf{u} = -\bar{U}^T \bar{U} \mathcal{L}(\mathbf{x}, \tau) \mathbf{x} \\ &= -\mathcal{L}(\mathbf{x}, \tau) \bar{U}^T \bar{U} \mathbf{x} \\ &= -\mathcal{L}(\mathbf{x}, \tau) \mathbf{x}_M \\ &= -\mathcal{L}(\mathbf{x}_M, \tau) \mathbf{x}_M - \underbrace{(\mathcal{L}(\mathbf{x}, \tau) - \mathcal{L}(\mathbf{x}_M, \tau)) \mathbf{x}_M}_{:=\mathbf{u}_p}, \end{aligned}$$

where the order of  $\bar{U}^T \bar{U}$  and  $\mathcal{L}$  can be swapped since the order of cyclic control and projection onto the subspace  $\mathcal{M}_n$  is irrelevant. Note that from the analysis conducted in Lemma IX.4, the center control plays no part in the convergence analysis to  $\mathcal{M}_{n\rho}$ . Thus we see that the dynamics for  $\mathbf{x}_M$  are perturbed by the term  $\mathbf{u}_p$  due to formation error. However since  $\|\mathbf{x} - \mathbf{x}_M\|$  is decreasing exponentially to 0, we have that  $\mathbf{u}_p$  is bounded and also decreasing exponentially to 0. Thus, there exists some time  $t^*$  following which the conditions given in Lemmas IX.4 and IX.5 are sufficient to ensure that  $\mathbf{x}_M \rightarrow \mathcal{M}_{n\rho}$ .  $\square$

**Remark IX.8** (Time Lag Parameter Bound). *Note that the bound for the time lag parameter used in the proof above (derived in the proofs for Lemmas IX.4 and IX.5) is less restrictive than the bound given in the statement of Theorem V.2 in the main body of the text. Thus, satisfaction of the constraint in (29) is sufficient for the convergence results to hold. We present the more restrictive condition in the main body of the text to simplify the exposition.*

Finally, we present the decoupled framework for controlling the formation size of a Johnson polyhedron by decoupling  $\mathcal{Q}_1$  from the set  $\mathcal{Q}$ . Suppose the tuple  $(\mathbf{n}_1, R_{\eta_1})$  describes the plane of the first polygon  $\mathcal{Q}_1$  and let  $(\mathbf{n}_2, R_{\eta_2})$  define the plane of an adjacent polygon  $\mathcal{Q}_2$  with robots  $j$  and  $j+1$  forming the shared edge. Define the variables  $\nu_f, \nu_b$  and  $k_r$  as follows:

$$k_r = (\mathbf{x}_{j+1} - \mathbf{x}_j) \cdot R_{\eta_2}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (60)$$

$$\nu_f = \begin{cases} 1 & \text{if } k_r > 0, \\ 0 & \text{else.} \end{cases} \quad (61)$$

$$\nu_b = \begin{cases} 1 & \text{if } k_r < 0, \\ 0 & \text{else.} \end{cases} \quad (62)$$

Then, the control law for robot  $i$  in  $\mathcal{Q}_1$  is given by:

$$\begin{aligned} \mathbf{u}_i^{(1)} &= \sum_{m=1}^{N_1} k_m^{(1)} [R_{m_s}^{(1)} (\mathbf{x}_{i+m}^{(1)} - \mathbf{x}_i^{(1)}) + R_{m_s}^{(1)T} (\mathbf{x}_{i-m}^{(1)} - \mathbf{x}_i^{(1)})] \\ &\quad + |k_r| [\nu_f R_{r_s}^{(1)} (\mathbf{x}_{i+1}^{(1)} - \mathbf{x}_i^{(1)}) + \nu_b R_{r_s}^{(1)T} (\mathbf{x}_{i-1}^{(1)} - \mathbf{x}_i^{(1)})], \end{aligned} \quad (63)$$

where  $R_{m_s} = R_{\eta_1}^T R_{m\pi/|\mathcal{V}_1| + \alpha_s(\mathbf{x}^{(1)})} R_{\eta_1}$ , and  $R_{r_s} = R_{\eta_1}^T R_{\pi/|\mathcal{V}_1|} R_{\eta_1}$ . Thus, the control for all robots in  $\mathcal{Q}_1$  is augmented by a “one-sided” cyclic controller (that is flow-invariant with respect to the polygon  $\mathcal{M}_{|\mathcal{V}_1|}^{(1)}$ ) that aligns  $\mathcal{Q}_1$  with its neighboring face  $\mathcal{Q}_2$ . Note that only one of  $\nu_f$  or  $\nu_b$  will be equal to 1 if  $\mathcal{Q}_1$  does not possess the correct in-plane rotation.

The proof of convergence for  $\mathbf{x}^{(1)}$  to the subspace  $\mathcal{M}_{|\mathcal{V}_1|}^{(1)}$  is a trivial extension of Lemma IX.4 by noting that the net control for  $\mathcal{Q}_1$  may be written as:

$$\dot{\mathbf{x}}^{(1)} = \mathbf{u}^{(1)} = -\mathcal{L}_{\eta}^{(1)}(\mathbf{x}^{(1)}, \tau)\mathbf{x}^{(1)} - |k_r(\mathbf{x}^{(1)})|(\nu_f L_1 \otimes R_{r_s} + \nu_b L_1^T \otimes R_{r_s}^T)\mathbf{x}^{(1)}.$$

A suitable auxiliary system for the above dynamics is given by:

$$\begin{aligned} \dot{\mathbf{y}}^{(1)} = & -\bar{V}^{(1)} \left[ \mathcal{L}_{\eta}^{(1)}(\mathbf{x}_p^{(1)}, \tau) \right. \\ & \left. + |k_r(\mathbf{x}_p^{(1)})|(\nu_f L_1 \otimes R_{r_s} + \nu_b L_1^T \otimes R_{r_s}^T) \right] \\ & \times \left( \bar{V}^{(1)T} \mathbf{y}^{(1)} + \bar{U}^{(1)T} \bar{U}^{(1)} \mathbf{x}_p^{(1)} \right), \end{aligned}$$

where  $\bar{V}^{(1)}$  is the orthonormal counterpart of  $\tilde{V}^{(1)}$  without the matrix  $E^{(1)}$ . Application of Corollary II.3 then requires:

$$\bar{V}^{(1)} \left[ \mathcal{L}_{\eta}^{(1)}(\mathbf{x}_p^{(1)}, \tau) + \frac{1}{2} |k_r(\mathbf{x}_p^{(1)})| \mathcal{L}_{1\eta}^{(1)} \right] \bar{V}^{(1)T} \succ 0, \text{ uniformly, } \forall \mathbf{x}_p, \tau.$$

Note that  $\bar{V}^{(1)} |k_r(\mathbf{x}_p^{(1)})| \mathcal{L}_{1\eta}^{(1)} \bar{V}^{(1)T} \succ 0, \forall \mathbf{x}_p$ . Thus, the above condition reduces to (55) in Lemma IX.2. The proof of convergence for polygon size is unchanged since the rotational controller introduced in (63) does not affect the expression for  $\dot{p}_i$  given in (57).

The control law for the remaining robots in the polyhedron is given as before by (20) and (21). Conditions for convergence for the reduced PPS (defined by the set of polygons  $\mathcal{Q} \setminus \mathcal{Q}_1$ ) can be derived by eliminating the first block in the net constraint matrix given in (17). The robots in  $\mathcal{V}_1$  converge to the polygon  $\mathcal{Q}_1$  with the desired size and an orientation consistent with the reduced PPS (as dictated by  $\mathcal{Q}_2$ ). In turn, all polygons in the reduced PPS will then possess the desired size  $\rho$  as dictated by the polygon  $\mathcal{Q}_1$ .

## X. QUADCOPTER VELOCITY TRACKING CONTROL

In this section we present the quadcopter velocity tracking controller used for the numerical simulations presented in the main body of the paper. For notational convenience we drop the  $i$  subscript notation referring to the  $i^{th}$  quadcopter. In this section we assume that all quantities refer to a single quadcopter.

Let  $\mathbf{v}_d$  denote the desired velocity for a quadcopter, as dictated by the cyclic control laws, and let  $\mathbf{v}$  be the current velocity. A simple proportional-integral-derivative controller converts the velocity error  $\mathbf{e}_v$  given by  $\mathbf{v}_d - \mathbf{v}$  into a desired acceleration  $\mathbf{a}_d$ . To achieve this desired acceleration, a reference attitude quaternion  $\mathbf{q}_d$  must also be generated since the quadcopter only possesses thrust capability in one direction (opposite its body  $z$  axis). To do this, we equate the desired acceleration to the actual acceleration dynamics given in (38) and linearize the resulting equation assuming small pitch and roll angles. This yields the following system of equations:

$$\begin{bmatrix} -T \sin(\psi) & -T \cos(\psi) \\ T \cos(\psi) & -T \sin(\psi) \end{bmatrix} \begin{bmatrix} \phi \\ \theta \end{bmatrix} = m_q \begin{bmatrix} a_{dx} \\ a_{dy} \end{bmatrix} \quad (64a)$$

$$-T + m_q g = m a_{dz}, \quad (64b)$$

where  $T$  is the desired net thrust, and  $[a_{dx}, a_{dy}, a_{dz}]$  are the components of  $\mathbf{a}_d$ . Given a desired yaw angle  $\psi$ , we can solve the above system of equations for the required thrust ( $T$ ), roll ( $\phi$ ) and pitch ( $\theta$ ). The roll, pitch and yaw angles are then converted into a desired attitude quaternion  $\mathbf{q}_d$  which is fed into the attitude control loop.

The attitude control loop uses the desired attitude  $\mathbf{q}_d$  and the current measured attitude  $\mathbf{q}$  to compute a desired correction quaternion  $\Delta \mathbf{q}$  given by:

$$\begin{aligned} \Delta \mathbf{q} &= \mathbf{q}_d \odot \mathbf{q}^{-1} \\ &= [\cos(\beta/2) \quad \sin(\beta/2)\hat{\mathbf{n}}_x \quad \sin(\beta/2)\hat{\mathbf{n}}_y \quad \sin(\beta/2)\hat{\mathbf{n}}_z]^T. \end{aligned}$$

The desired moment is then given by the following proportional-derivative control law:

$$\mathbf{M} = K_p \beta \hat{\mathbf{n}} - K_d \boldsymbol{\omega},$$

where  $K_p$  and  $K_d$  are positive semi-definite gain matrices. Finally, the desired thrust and moment are fed into the quadcopter motor controller to compute the desired motor speeds. A block diagram for this controller is shown below in Figure 11.

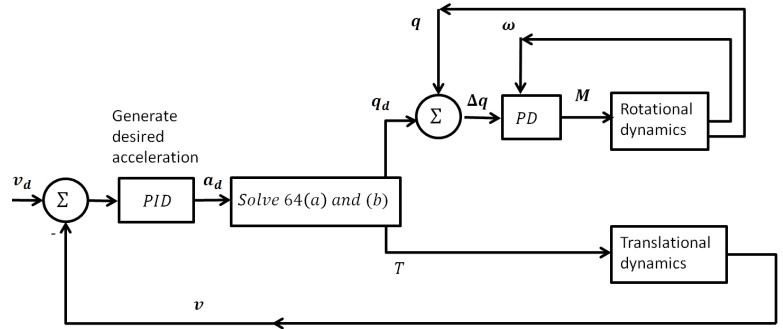


Fig. 11: Quadcopter Velocity Tracking Controller.